

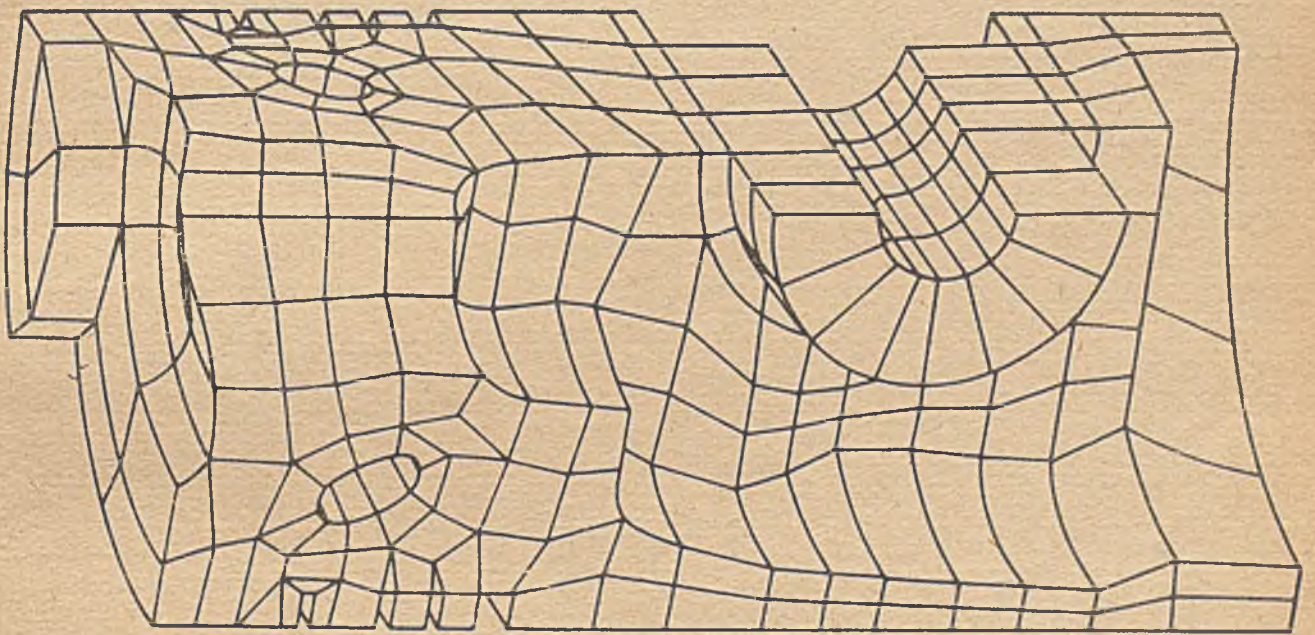
biuletyn informacyjny

P.3057/79

2
'79



OBIEKTOWE
SYSTEMY
KOMPUTEROWE



Zjednoczenie Przemysłu Automatyki i Aparatury Pomiarowej „MERA”
Instytut Maszyn Matematycznych „MERA IMM” Branżowy Ośrodek INTE

Rysunek na okładce: Model tłoka silnika Diesla. Przygotowanie takiego modelu zajęłoby wykwalifikowanemu inżynierowi około 4 tygodni. Zastosowanie interakcyjnych systemów graficznych i generowania automatycznego skraca czas wykonania modelu do kilku dni.
/Wg Computer Aided Design 1978 nr 2/

Druk okładki IMM zam. 112/79 nakł. 820 egz.

Biuletyn Informacyjny OBIEKTOWE SYSTEMY KOMPUTEROWE

Rok XVII

Nr 2

1979

P. 3057/79

Spis treści

Содержание

Contents

JAGIELSKI J. - Systemy komputerowego sterowania procesami wytwarzania w pewnej klasie obiektów s. 3

KEDZIOR Z. - Laserowe urządzenia do automatyzacji korekcji rezystorów warstwowych .. s. 36

WRONA J. - Organizacja bezpośredniego wprowadzania informacji w obiektowych systemach sterowania s. 45

ROSZKOWSKA E. - Dokumentowanie oprogramowania EMC s. 50

MITAL Z., SZEJKO S. - Zagadnienia automatyzacji badań silników spalinowych za pomocą dwuprocesorowego zestawu minikomputerów serii MERA-300 .. s. 57

MANKIEWICZ J., SZEJKO S., WIERNIEWICZ J. - Wielodostępny system symulacji minikomputerów MONIK 86/100 s. 62

KOWALEWSKI L., KUCHCIŃSKI K., WISZNIEWSKI B. - System wielomaszynowy oparty na minikomputerze z serii MERA-300 .. s. 65

KOWALEWSKI L., KUCHCIŃSKI K., WISZNIEWSKI B. - Niektóre problemy rekonfiguracji systemów wielomaszynowych zbudowanych na maszynach cyfrowych serii MERA-300 s. 72

MURASZKIEWICZ M. - Realizacja mnożenia, wyszukiwania i sortowania za pomocą układów kombinacyjnych s. 79

Ягельски Е.- Системы компьютерного управления изготовительными процессами в некотором классе объектов с. 3

Кендзёр З.- Лазерные устройства для автоматической корректировки непровольных регистров с. 36

Врона Я.- Организация непосредственного ввода информации в объектных системах управления с. 45

Ротковска Е.- Документирование программного обеспечения ЭВМ с. 50

Миталь З.,Шейко С.- Вопросы автоматизации испытаний двигателей внутреннего сгорания с помощью двухпроцессорного блока мини ЭВМ серии МЭРА-300 с. 57

Манкевич Е.,Шейко С., Веремвич Я.- Многодоступная система имитации мини ЭВМ Молик 86/100 с. 62

Ковалевски Л., Кухчиньски К., Вишневи Б.- Многомашинная система, основанная на мини ЭВМ серии МЭРА-300 с. 65

Ковалевски Л., Кухчиньски К., Вишневи Б.- Некоторые проблемы реконфигурации многомашинных систем, реализованных на ЭВМ серии МЭРА-300 с. 72

Мурашкевич М.- Реализация умножения, поиска и сортирования с помощью ячеечных схем ... с. 79

JAGIELSKI J. - Computer control systems of production processes for a given class of objects .. s. 3

KEDZIOR Z. - Laser devices for automatic laminar resistor correction s. 36

WRONA J. - Organization of information input to computer remote control system devices ... s. 45

ROSZKOWSKA E. - Computer software documentation s. 50

MITAL Z., SZEJKO S. - Automation problems of internal combustion engine researches using the MERA-300 two-process minicomputer set s. 57

MANKIEWICZ J., SZEJKO S., WIERNIEWICZ J. - Multiaccess simulation system of the MONIK 86/100 minicomputers s. 62

KOWALEWSKI L., KUCHCIŃSKI K., WISZNIEWSKI B. - Multimachine system based on the MERA-300 minicomputer s. 65

KOWALEWSKI L., KUCHCIŃSKI K., WISZNIEWSKI B. - Some reconfiguration problems of multimachine systems built on the MERA-300 digital machines s. 72

MURASZKIEWICZ M. - Realization of multiplication, searching and sorting using cellular systems s. 79

D W U M I E S I Ę C Z N I K

Wydaje:

CENTRUM NAUKOWO-PRODUKCYJNE TECHNIK KOMPUTEROWYCH I POMIARÓW
I N S T Y T U T M A S Z Y N M A T E M A T Y C Z N Y C H
Branżowy Ośrodek Informacji Naukowej Technicznej i Ekonomicznej

KOMITET REDAKCYJNY

dr inż. Stanisława BONKOWICZ-SITTAUER, mgr Hanna DROZDOWSKA
/sekretarz redakcji/, dr inż. Marek HOLYŃSKI,
doc.dr inż. Henryk ORŁOWSKI /redaktor naczelny/,
mgr inż. Jerzy MYSIOR, mgr inż. Józef SZMYD, mgr Robert ZAJĄC

Opracowanie graficzne: Barbara KOSTRZEWSKA

Adres redakcji: ul. Krzywickiego 34, 02-078 Warszawa
tel. 21-84-41 w. 431 lub 28-37-29

Systemy komputerowego sterowania procesami wytwarzania w pewnej klasie obiektów

Cz.1. Niektóre zagadnienia projektowania

Wstęp

Coraz częściej zachodzi konieczność komputeryzacji wielu czynności nadzoru i sterowania procesu wytwarzania. Należy przyjąć, że docelowo większość tych zadań powinna być oparta na krajowej bazie badawczo-projektowej i na krajowym sprzęcie. Stąd wynika konieczność wypracowania metod prowadzenia prac przy projektowaniu systemów komputerowego sterowania (SKS) obiektami w zakładach przemysłowych. Celowe może być podzielenie się doświadczeniami w zakresie projektowania SKS, tzn. omówienie zagadnień, z którymi może spotkać się projektant pilotowych SKS. Omówione w artykule zagadnienia są próbą uogólnienia wniosków i spostrzeżeń z prowadzonych w Instytucie Maszyn Matematycznych prac przy projektowaniu SKS, a głównie systemu komputerowego sterowania centrum obrabiarek* - SKSC-1. Ze względu na ramy tego artykułu omówiono tylko niektóre zagadnienia dotyczące początkowych etapów projektowania.

Chociaż każdy typ obiektu-procesu oraz inne warunki narzucają projektantowi SKS określone wymagania i ograniczenia, jednak wiele zagadnień jest wspólnych (ogólnych). Wydaje się więc, że artykuł może zainteresować projektantów SKS. W szczególności dotyczy to projektantów SKS dla obiektów w przemyśle maszynowym. Natomiast projektantów obiektów może zainteresować z punktu widzenia - jak projektant SKS "widzi" obiekt.

Ilustracją omawianych zagadnień jest opis CP-KOR-1 (druga część artykułu) oraz sterujący nim SKSC-1.

Charakterystyka klasy obiektów i klasy SKS

Z punktu widzenia projektanta SKS znaczna część obiektów w przemyśle maszynowym charakteryzuje się następującymi istotnymi cechami:

- jest to zbiór jednostek funkcjonalnych, które odpowiednio sterowane przez SKS realizują określone fragmenty procesu produkcyjnego;
- praca tych jednostek jest w pewnym zakresie wzajemnie niezależna, może zachodzić jednak konieczność skoordynowania przez SKS pracy pewnych jednostek;
- przebiegający w obiekcie proces oraz wyroby mają charakter dyskretny;
- pewien zakres automatyzacji urządzeń obiektu sprawia, że sygnały o zmianach ich stanu (zdarzeniach) mają charakter binarny i wymagają one (urządzenia) na ogół binarnego sterowania;
- w obiekcie występują częste zmiany procesu wynikające ze zmiany zadań, zakłóceń, maloweryjnej produkcji itp.;
- proces (i urządzenia) nie stawia w zasadzie krytycznych wymagań na czas reakcji SKS na zachodzące w obiekcie zdarzenia i opóźnienie reakcji nie powoduje awarii.

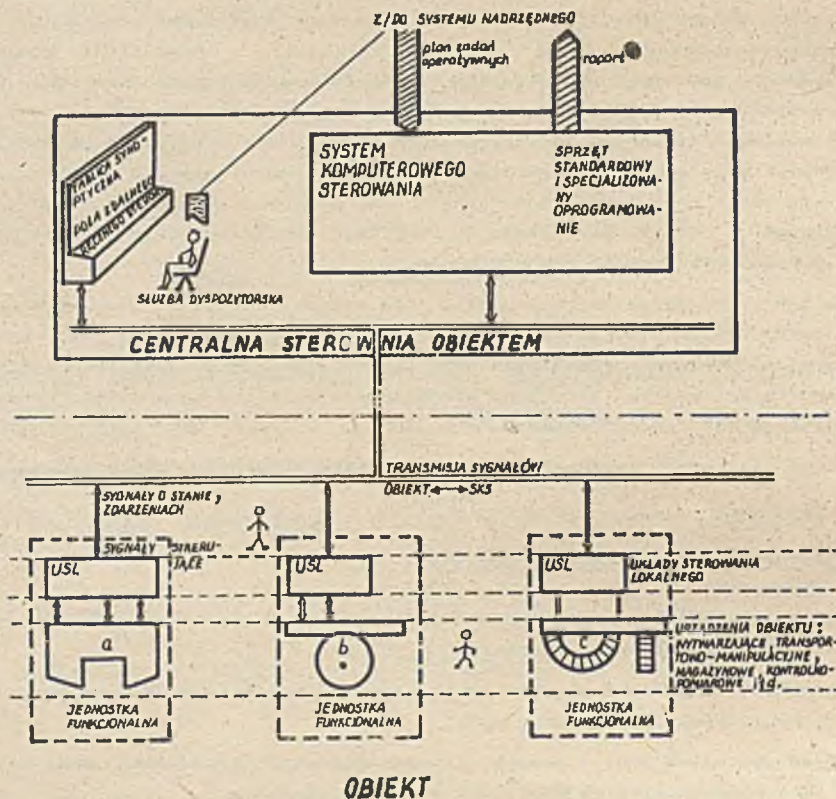
Powyższe cechy obiektu w znacznym stopniu określają cechy SKS. Są to między innymi:

- przyjmowanie i analizowanie (przez SKS) stosunkowo dużej liczby sygnałów o stanie obiektu;

* Centrum produkcyjne obrabiarek CP-KOR-1 opracowano i zbudowano w Centrum Badawczo-Konstrukcyjnym Obrabiarek (CBKO) w Pruszkowie.

- duża liczba sytuacji w obiekcie, wymagających podejmowania przez SKS różnorodnych decyzji z ustalonego zbioru decyzji (np. na podstawie tablic decyzyjnych);
- konieczność "współpracy" z obsługą obiektu;
- konieczność sterowania przy zmianach konfiguracji obiektu w sytuacjach szczególnych (np. uszkodzenie jakiegoś urządzenia);
- konieczność zapewnienia szybkiego i prostego "startu" i "restartu" SKS;
- zapewnienie właściwej niezawodności systemowej (w tym niezawodności sprzętu).

Przykład obiektu i jego powiązania z SKS podaje rys. 1.



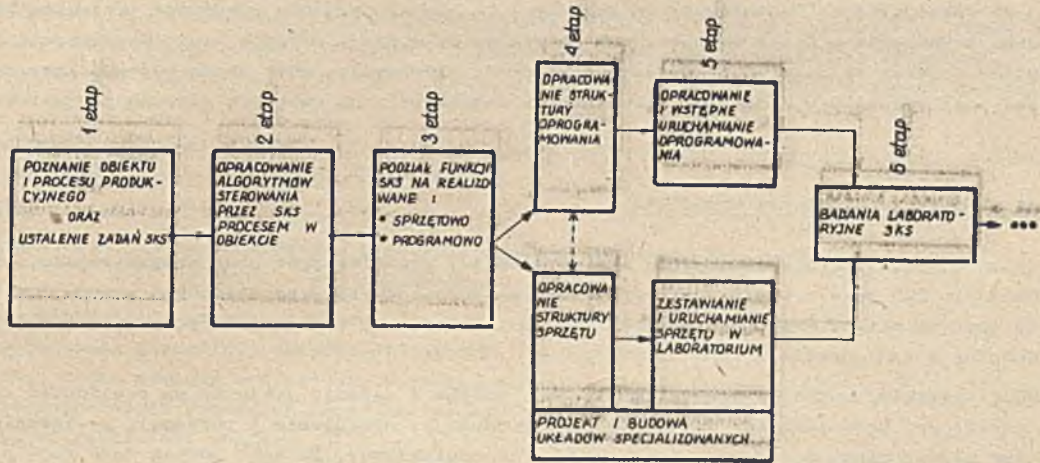
Rys. 1. Schemat blokowy powiązań (sprzężeń) obiekt - SKS

W urządzeniach a, b, c obiektu, w trakcie procesu produkcyjnego, zachodzą określone zdarzenia. Na pewne zdarzenia zareagują układy sterowania lokalnego USŁ, np. posyłając odpowiednie sygnały sterujące do urządzenia. O innych zdarzeniach w urządzeniu USŁ informowany jest SKS. Po analizie powstałej sytuacji (często nie tylko w danym urządzeniu) SKS powinien odpowiednio zareagować na te zdarzenia. Celem jest informowanie na bieżąco dyspozytora o sytuacji w obiekcie (tablica synoptyczna) oraz zapewnienie mu możliwości (np. w sytuacjach awaryjnych) przejęcia sterowania (pola zdalnego ręcznego sterowania).

Etapy prac przy opracowywaniu projektu SKS

Projektowanie SKS, podobnie jak inne przedsięwzięcia techniczne, wymaga ustalenia właściwej kolejności prac. Przykład podziału na etapy początkowej fazy projektowania pilotowego SKS podano na rys. 2. Dla innych przypadków etapy te mogą być nieco inne. Pracochłonność w ramach poszczegól-

nych etapów zależy głównie od wielkości, złożoności obiektu i przygotowania-doświadczenia zespołów projektanta SKS i projektanta obiektu.



Rys. 2. Etapy projektowania pilotowego SKS

Obiekt i przebiegający w nim proces produkcyjny może być różnie (dla różnych potrzeb) scharakteryzowany. Konieczna jest też charakterystyka obiektu i procesu z punktu widzenia projektanta SKS, gdyż to pozwoli na ustalenie zadań SKS w zakresie sterowania. Dla ustalenia całości zadań SKS należy także określić jakie i w jakiej formie będą do SKS dostarczane dane o planowanych dla obiektu zadaniach produkcyjnych. Opracowanie tych zagadnień zamyka pierwszy etap prac: poznania obiektu i określenie zadań SKS w zakresie sterowania.

Kolejnym etapem jest opracowanie algorytmów sterowania obiektem. Na tym etapie - na podstawie wyników prac pierwszego etapu - powinny być opracowane algorytmy sterowania przez SKS procesem w obiekcie. Algorytmy te powinny dostatecznie dokładnie określać jak SKS zareaguje na zdarzenia-stany w danym urządzeniu obiektu, z uwzględnieniem koniecznej koordynacji działania urządzenia jako części składowej obiektu.

W trzecim etapie powinno się dokonać podziału zadań-funkcji SKS na realizowane programowo (software*), na realizowane w specjalistycznym sprzęcie oraz na realizowane przez obsługę.

Podział ten pozwala w czwartym etapie dobrać odpowiednią strukturę sprzętu i odpowiednią strukturę oprogramowania. Struktura sprzętu ma oczywisty wpływ na strukturę oprogramowania. Można wykazać, że struktura oprogramowania wpływa też na strukturę sprzętu. Dlatego na tym etapie konieczna jest ścisła współpraca projektantów obu struktur, co pozwoli na możliwie optymalny dobór tych struktur.

Dalsze etapy - nie omawiane w tym artykule - to równoległe prowadzone prace przy kompletowaniu sprzętu i opracowywaniu - (adaptowaniu) odpowiedniego oprogramowania. Etap ten wymaga odpowiednich zestawów sprzętu dla uruchamiania programów. Jednocześnie powinno się rozpocząć opracowywanie metod uruchamiania i badania SKS (sprzętu i oprogramowania) w warunkach laboratoryjnych z symulacją obiektu, gdyż nie zawsze jest możliwe i celowe uruchamianie SKS od razu we współpracy z obiektem. Po uruchomieniu i przebadaniu SKS w warunkach laboratoryjnych oraz po włączeniu ewentualnych poprawek, przystąpić można do uruchomienia i badań SKS z obiektem. Po tym następuje ostatni etap eksploatacji próbnej i szkolenia obsługi. Do tego czasu powinna być przygotowana i zweryfikowana dokumentacja techniczno-ruchowa. Do prowadzenia prac w ramach ostatnich dwóch etapów konieczny jest odpowiednio przygotowany obiekt, np. powinna być zapewniona możliwość sprawdzo-

* Rozumiane tu jako realizowane przez zestawy komputerów

nia SKS w określonych sytuacjach szczególnych - awaryjnych. Wymaga to ustalenia trybu prac - wspólnego z projektantem obiektu-użytkownikiem.

Często, z różnych przyczyn, konieczne są uściślenia, uzupełnienia lub korekty ustaleń z poprzednich etapów. Trzeba więc ustalić tryb merytoryczny rozpatrywania (analizy) tych zmian i wnoszenia ich do dokumentacji. Podyktowane to jest tym, że niejednokrotnie prace nad projektem SKS są prowadzone równolegle w kilku zespołach, a ustalenia mają swoje odbicie w wielu dokumentach. Szczególnie dotyczy to ustaleń w zakresie algorytmów sterowania, gdyż muszą być one korygowane nie tylko przy zmianach w obiekcie, ale często w trakcie prac na dalszych etapach projektowania SKS.

Poznanie obiektu i określenie zadań SKS

Poznanie obiektu, mające zasadniczy wpływ na dalszy przebieg prac przy projektowaniu, budowie i uruchamianiu SKS jest z wielu przyczyn jednym z trudniejszych, czasochłonnych etapów, szczególnie, gdy jest to obiekt większy, bardziej złożony. Składa się na to wiele przyczyn, z których część omówimy w tym punkcie.

W chwili przystępowania do projektowania SKS, obiekt i zadania jakie ma on realizować są w zasadzie określone. Określone są: zestaw środków produkcji, urządzenia i personel, powiązania technologiczne między nimi, znany jest także proces technologiczny. Na ogół jednak dane to - przygotowane przez projektanta obiektu - są niewystarczające dla projektanta SKS ani co do treści ani co do formy. Ponadto często urządzenia obiektu (i proces produkcyjny) nie są przystosowane do sterowania z SKS i muszą być zmodyfikowane oraz uzupełnione warstwą układów sterowania lokalnego USL. Także pogląd projektanta obiektu dotyczący zakresu zadań SKS może wymagać weryfikacji. Nie zawsze jest uzasadnione obciążenie SKS sterowaniem urządzeniami we wszystkich możliwych stanach, np. dla zdarzeń, które występują stosunkowo rzadko może wystarczyć powiadomienie przez SKS obsługi, która odpowiednio zareaguje na to zdarzenie. Natomiast uzasadnione może okazać się obciążenie SKS funkcjami ciągłego nadzoru nawet jednostek o dużej autonomii - np. ciągle sprawdzanie czy przetwarzane prefabrykaty opuszczają jednostkę z określonym rytmem.

Ponieważ na tym etapie trudno jest określić jakie obciążenie zadaniami SKS jest uzasadnione technicznie i ekonomicznie, celowe jest zadania te podzielić na: konieczne, pożądane i dodatkowe. Dwie ostatnie grupy zadań będą uwzględnione w miarę posiadanych rezerw w SKS.

W omawianej klasie celem obiektu jest wytwarzanie określonych wyrobów z dostarczanych surowców, półfabrykatów. Jakie i w jakich ilościach mają być wytwarzane wyroby określa plan operatywny dla obiektu.

Wytwarzanie wyrobu wymaga określonego procesu produkcyjnego, a to z kolei wymaga odpowiednich urządzeń i personelu mogących wykonywać przewidziane w procesie czynności i zabiegi technologiczne. Urządzenia i personel są wzajemnie powiązane, tworzą strukturę obiektu. Są one (urządzenia i personel) w określonym zakresie samosterowne, a w określonym zakresie są podatne na sterowanie zewnętrzne tzn. wymagają sterowania zewnętrznego (np. z SKS). Przez uzupełnienie urządzeń odpowiednimi układami sterowania lokalnego USL można lepiej przystosować te urządzenia do sterowania z SKS.

Łącznie obiekt i SKS powinny zapewnić właściwy przebieg procesu produkcyjnego tak, aby były realizowane - wyznaczone planem operatywnym zadania.

Dla scharakteryzowania obiektu konieczna jest znajomość, między innymi, następujących jego cech:

- zakresu samosterowności (autonomii) i zakresu podatności (wymagań) na sterowanie zewnętrzne poszczególnych jednostek funkcjonalnych (urządzeń uzupełnionych USL, obsługi);
- procesów produkcyjnych dla poszczególnych typów wyrobów (tzn. jakim zabiegom technologicznym, w jakiej kolejności i w jakich jednostkach funkcjonalnych ma być poddany obróbee półfabrykat aby otrzymać żądany wyrób);
- struktury obiektu.

Opisy tych cech muszą być dostatecznie precyzyjne i jednoznaczne, gdyż - jak wspomniano - będą one zasadniczym dokumentem określającym zadania SKS.

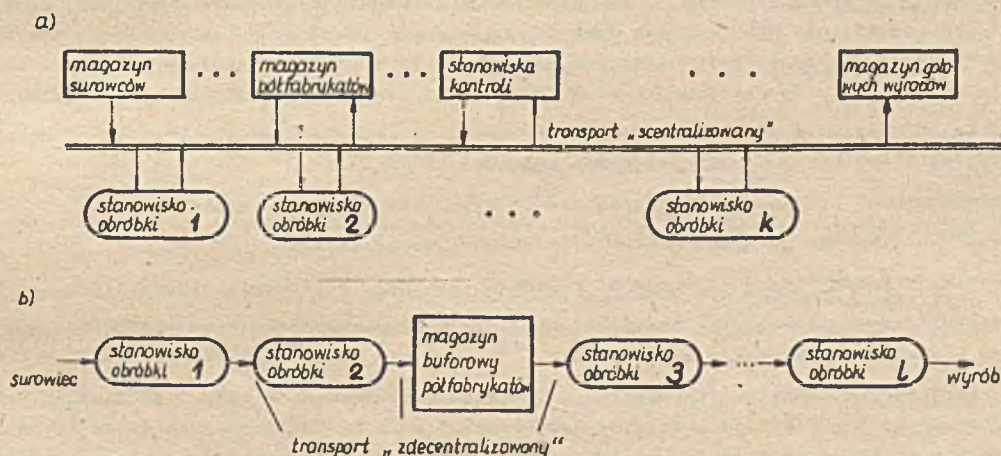
Pierwszą, z wyżej wymienionych cech można opisać bądź w formie schematów sieci działań (flow-diagramy), bądź w formie sieci Petri, ewentualnie łącząc odpowiednio te dwie formy.

Sieć uzupełniona przewidywanymi czasami wykonywania czynności ($t = \min \dots \max \dots$), oraz danymi o postaci sygnałów sterujących i sygnałów o stanie, opisuje zasadę działania urządzenia i jego współdziałanie z otoczeniem (wymagania dotyczące sterowania i koordynacji z innymi urządzeniami).

Z opisu procesu produkcyjnego - oprócz wymienionych już kolejności zabiegów technologicznych - powinny wynikać:

- charakter produkcji: - taśmowa, gniazdowa
 - jednoczesna produkcja jednego lub wielu asortymentów
 - wielko- lub małoseryjna
- warunki w obiekcie rozpoczęcia procesu;
- warunki (w danej jednostce) rozpoczęcia obróbki prefabrykatu;
- działania, które powinny być podjęte w razie zakłóceń procesu produkcyjnego (wpływ zakłócenia na pracę pozostałych jednostek funkcjonalnych);
- kryteria optymalizacji procesu.

W omawianej klasie większość struktur obiektów da się sprowadzić do jednej z przedstawionych na rys. 3. Najczęściej organizacja przepływu-transportu półfabrykatów między jednostkami przetwarzającymi określa strukturę obiektu (taśma, system gniazdowy itp.). To z kolei narzuca wymagania dotyczące sterowania nie tylko urządzeniami transportu, ale także wpływa na wymagania dotyczące odpowiedniej koordynacji synchronizacji pracy jednostek przetwarzających. W strukturze mogą też wystąpić odpowiednie magazyny buforowe łagodzące wymagania dotyczące koordynacji. Na rys. 3 przedstawiono struktury transportu scentralizowanego i zdecentralizowanego.



Rys. 3. Przykłady struktur transportu

Podstawowymi jednostkami w obiekcie są jednostki przetwarzające, gdyż w nich przebiega zasadnicza część procesu (obróbka, montaż itp.). Efektywność pracy całego obiektu w oczywisty sposób zależy od ich pracy. Pracę innych jednostek obiektu ocenia się na podstawie tego jak zapewniają właściwą pracę jednostek przetwarzających.

Omówione wyżej cechy obiektu i procesu pozwalają na określenie zadań SKS w zakresie bezpośredniego sterowania i nadzoru czynnościami w urządzeniach obiektu. Ponieważ jednak SKS ma tak sterować procesem, aby w obiekcie były realizowane określone dla obiektu planowane zadania należy ustalić, jak i przez kogo, będzie opracowywany plan operatywny zadań dla obiektu. Należy ustalić jaki

pozostawia się stopień swobody dla SKS przy ewentualnej korekcie-adaptacji planu do zmian sytuacji w obiekcie.

Jak już wyżej stwierdzono ogólnie, podstawowym zadaniem SKS jest takie reagowanie na zdarzenia w obiekcie, aby przy danych warunkach były realizowane zadania obiektu tzn. aby odpowiednio przebiegał proces produkcyjny. W zadaniach SKS znajdują się:

- bezpośredni nadzór i sterowanie sekwencjami czynności w urządzeniu-jednostce funkcjonalnej tak, aby realizowało ono odpowiedni (w danych warunkach) fragment procesu produkcyjnego,
- wyznaczanie następnego bieżącego zadania jednostki po zakończeniu przez nią poprzedniego oraz przygotowanie startu jednostki do realizacji tego zadania (np. zlecenie dostarczenia odpowiednich prefabrykatów, ustawienie stanu startowego),
- wyznaczanie bieżących zadań całego obiektu na podstawie planu operatywnego i aktualnego stanu obiektu, a w razie zakłóceń-adaptacji tych zadań do zmienionego stanu,
- przyjmowanie z systemu nadrzędnego planu zadań operatywnych i odpowiednie ich przetwarzanie,
- przygotowanie dla systemu nadrzędnego i obsługi raportów o przebiegu procesu.

Takie, lub inne - bardziej dopasowane do konkretnego obiektu - pogrupowanie zadań pozwala na dekompozycję złożonego zadania sterowania obiektem. Ułatwi to - w następnych etapach - opracowanie algorytmów, programów i uruchamianie SKS.

Ilustracją omawianych zagadnień jest opis CP KOR-1, jako obiektu sterowanego przez SKSC-1, zamieszczony w części II.

Jak wynika z rys. 1 między urządzeniami obiektu, a SKS (ściślej transmisją sygnałów) występuje warstwa układów sterowania lokalnego USL. Warstwa ta w większym lub mniejszym stopniu separuje SKS od bezpośredniego sterowania i śledzenia elementarnych czynności (mikroczynności) układów wykonawczych w urządzeniach obiektu. Pogłębianie (rozbudowa) tej warstwy prowadzi do zwiększenia autonomii urządzeń jednostek obiektu, a w konsekwencji do zmniejszenia zbioru zdarzeń, na które ma reagować SKS (odciążenie SKS). Jednak zbytne pogłębianie prowadzi do zmniejszenia wpływu SKS na proces w obiekcie (zmniejszenia podatności na sterowanie z SKS). Często też rozbudowa USL jest ekonomicznie uzasadniona. Oprócz przejęcia części funkcji sterowania, USL mają za zadanie:

- dopasowanie pod względem technicznym (elektrycznym i czasowym), "sygnałów z urządzeń obiektu do założonego w SKS (transmisji) standardu sygnałów
- ułatwienie obsłudze przygotowania stanów wyjściowych (startowych) urządzeń,
- umożliwienie-ułatwienie lokalnej kontroli sprawności urządzeń,
- umożliwienie, w szczególnych sytuacjach, ręcznego lokalnego sterowania urządzeniami,
- wraz z odpowiednimi mechanizmami urządzeń, utworzenie odpowiedniej blokady zapobiegającej uszkodzeniom w razie niewłaściwego wystrojenia.

Ogólnie rzecz biorąc, warstwa ta zwiększa autonomię (samosterowność) urządzeń obiektu oraz ułatwia obsługę urządzeń. Dobór odpowiedniej "głębokości" warstwy USL zależy od wielu czynników, przede wszystkim od urządzeń obiektu, procesu produkcyjnego i założonej autonomii jednostek funkcjonalnych. Ponieważ USL najczęściej są umieszczane w hali obiektu - konstrukcja ich powinna spełniać odpowiednie wymagania mechaniczno-klimatyczne i ergonomiczne. Stąd wynika, że określenie USL jest złożonym problemem. Przy tym powinno się tego dokonać dostatecznie wcześnie, gdyż w bezpośredni sposób rzutuje to na dalsze prace zarówno u projektanta SKS jak i projektanta obiektu. Często ze względu na lepszą znajomość samych urządzeń oraz innych warunków celowe jest - po ustaleniu założeń na USL - powierzenie zaprojektowania i budowy USL grupie projektanta obiektu.

Po ustaleniu warstwy USL, urządzenia obiektu i czynności, które w nich są wykonywane w trakcie procesu, będą inaczej "widziane" przez SKS (separacja), co powinno być uwzględnione w opisach charakterystyk - cech obiektu.

Gdy bezpośredni nadzór i sterowanie jednostką funkcjonalną obiektu jest zadaniem złożonym, a przy tym wskazane jest zachowanie odpowiedniej autonomii, może okazać się celowe i ekonomicznie

uzasadniono zastosowanie mikrokomputera (sterownika mikroprocesorowego). Wówczas pojawia się zagadnienie, czy mikrokomputer zaliczyć do USL, czy do SKS (może tutaj decydować jego lokalizacja), a to wymaga innego podejścia do opisu obiektu, innego podziału zadań.

Opracowanie algorytmów sterowania

Prace przeprowadzone w ramach pierwszego etapu określiły czego oczekuje lub wymaga obiekt od SKS.

Jak już kilkakrotnie stwierdzono, urządzenia obiektu wymagają odpowiedniego - dla danej sytuacji w samym urządzeniu, jak i sytuacji w innych współpracujących z nim urządzeniach -ysterowania. Ysterowanie to powinno być też zgodne z bieżącym dla danego urządzenia zleceniem. Ze swojej strony urządzenia wykazują określoną podatność na sterowanie z SKS oraz określoną samosterowność i посыłają do SKS określony zbiór sygnałów o swych stanach-zdarzeniach.

W ramach prac drugiego etapu powinno się określić relacje między sygnalizowanymi stanami urządzeń-jednostek, a ich ysterowaniem (posyłaniem sygnałów sterujących przez SKS). Konieczne jest więc opracowanie algorytmów sterowania i algorytmów koordynowania.

Zależności, które występują w pracy jednostek-urządzeń realizujących proces produkcyjny narzucają hierarchiczny układ algorytmów.

Pierwszy poziom, to algorytmy sterowania czynnościami poszczególnych jednostek, w ramach otrzymanych z drugiego poziomu zleceń. Zaawansowanie realizacji zlecenia jest sygnalizowane do poziomowi drugiego z odpowiednią dyskretnością (kolejne kroki).

Poziom drugi i ewentualnie kilka wyższych to algorytmy koordynujące pracę między poszczególnymi jednostkami, przez przekazywanie odpowiednich i w odpowiednim czasie zleceń. Zlecenia te powinny być zgodne z bieżącymi zadaniami zespołu urządzeń lub obiektu.

Bieżące zadania obiektu są wypracowywane przez algorytmy najwyższego poziomu na podstawie operatywnego planu zadań przekazywanego do SKS z zewnątrz. Algorytmy te powinny określić, jak operatywny plan zadań będzie przetworzony na bieżące zadania obiektu lub zespołów urządzeń. Przy tym powinno być ustalone, jaki stopień swobody pozostawia się SKS (algorytmom tego poziomu) w adaptowaniu tego planu do aktualnej sytuacji w obiekcie.

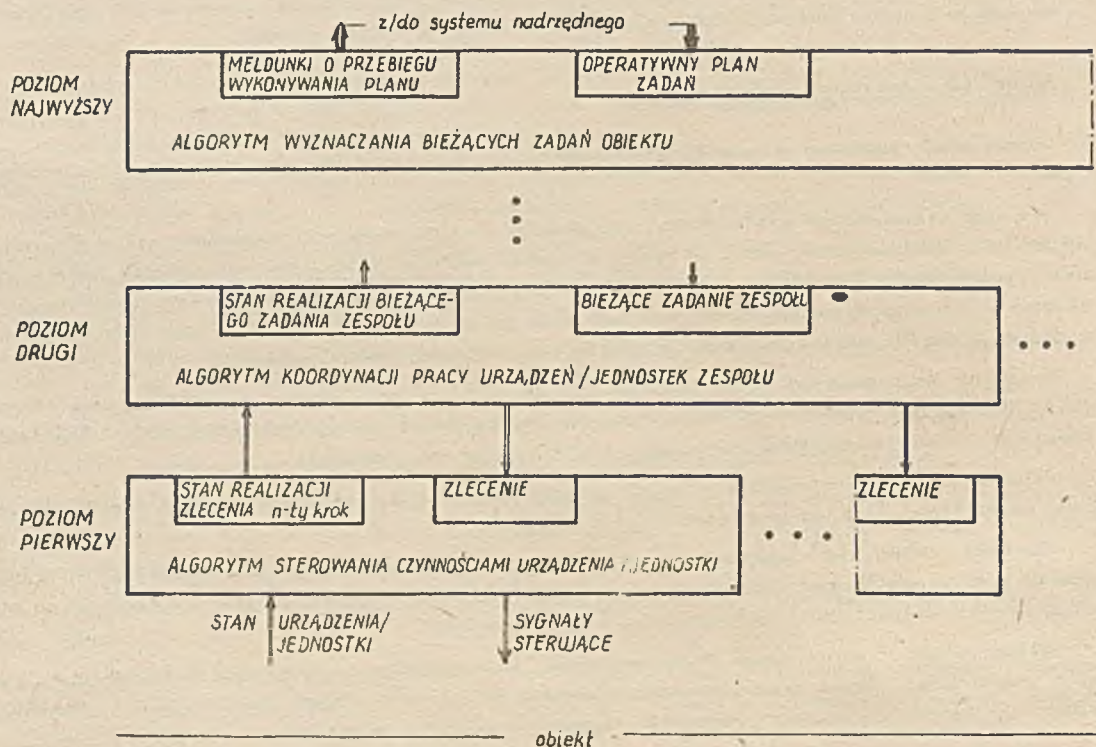
Liczba poziomów i podział pionowy powinien być odpowiednio dobrany do zadań SKS. Zwiększenie liczby poziomów i pionów z jednej strony wymaga uwzględnienia w algorytmach konieczności wzajemnego przekazywania danych, z drugiej strony uzyskany stopień modułowości może ułatwić - w następnych etapach - dobór odpowiedniej struktury sprzętu i oprogramowania. Ogólny schemat podziału algorytmów na poziomy podaje rys. 4.

Celowe jest opracowywanie algorytmów kolejno - poczynając od pierwszego poziomu. Podyktowane to jest tym, że opracowanie niższego poziomu pozwala określić jakich danych (zleceń, zadań) oczekuje ten poziom od wyższego.

W trakcie opracowywania algorytmów, powinno się dokonać analizy realizowalności wymagań na sterowanie stawianych przed SKS. W tym celu należy dokonać konfrontacji tych wymagań z podatnością i samosterownością urządzeń obiektu.

Często liczba stanów występujących w urządzeniach, które (dla uniknięcia nieokreśloności) muszą być uwzględnione w algorytmach - jest znaczna. Komplikuje to opracowywane algorytmy. Jednak tylko część z tych stanów może wystąpić przy normalnym przebiegu procesu i dla nich opracowuje się szczegółowe algorytmy. Wystąpienie stanu, który nie jest objęty tą częścią uznaje się jako sytuację szczególną (zakłócenie, awaria), a algorytm dla tych przypadków może być wspólny i ograniczony do powiadomienia obsługi (ewentualnie wyłączenia urządzenia).

W zasadzie dane ysterowanie powinno spowodować przejście urządzenia w ściśle określony stan (często jeden stan). Dlatego też wystąpienie - po danym ysterowaniu - innego, nieprzewidzianego stanu może być uznane za zakłócenie i spowodować np. powiadomienie obsługi. Uwzględnienie tego w algorytmach chroni w pewnym zakresie przed niewłaściwym ysterowaniem.

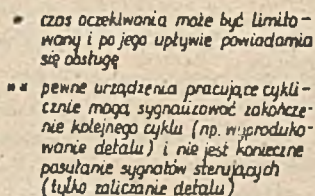


Rys. 4. Poziomy algorytmów

Algorytmy wykazują dużą różnorodność, gdyż algorytmy poziomu pierwszego muszą uwzględniać specyfikę urządzeń, a algorytmy wyższych poziomów strukturę obiektu, proces produkcyjny i zadania. Dlatego podanie ogólnej struktury algorytmów jest trudne. Podobnie trudno jest podać - poza ogólnie znanymi - zasady opracowywania algorytmów.

Na rys. 5 podano przykład algorytmu sterowania urządzeniem (poziom pierwszy). Po wysłaniu sygnałów sterujących i ich przyjęciu przez urządzenie oczekuje się na zmianę stanu urządzenia. Gdy urządzenie zasygnalizuje zmianę lub przejsze w inny stan, sprawdza się czy urządzenie przeszło w przewidywany stan. Gdy nie - powiadamia się obsługę, która powinna ustalić przyczynę zakłócenia i po jej usunięciu doprowadzić do restartu. Gdy tak - sprawdza się, czy spełnione są określone warunki dla kolejnego wystrojenia urządzenia. Mogą to być warunki, na spełnienie których należy tylko poczekać bez odwoływania się lub powiadamiania innych algorytmów. W innych wypadkach spełnienie warunków jest zależne od innych algorytmów np. od wystrojenia innego urządzenia. W tej sytuacji konieczne jest powiadomienie tych algorytmów. Gdy wszystkie warunki będą spełnione, do urządzenia posyła się odpowiednie sygnały sterujące i ustala się stan w jaki powinno przejść urządzenie po wykonaniu zadanych tymi sygnałami czynności.

Jak wynika z powyższego opisu, część warunków umożliwiających dalsze sterowanie urządzeniem może być spełniona tylko przy współdziałaniu z innymi algorytmami. Może to wymagać skoordynowania działania urządzeń sterowanych przez te algorytmy. Funkcje te najczęściej należą do algorytmów poziomu drugiego i są realizowane pośrednio przez odpowiednie i w odpowiednim czasie przekazywane zlecenia. Występują też sytuacje, gdy celowe może okazać się przypisanie tych funkcji algorytmom poziomu pierwszego. Wówczas sygnał o gotowości do współdziałania obu urządzeń może być bezpośrednio wprowadzony jako warunek dalszego sterowania i urządzenie, które wcześniej "doszło" do stanu gotowości "poczeka" na drugie. Są też sytuacje, gdy któreś z urządzeń nie może "czekać" - pracuje z własnym rytmem. Dlatego stany, w których nie wystąpi zsynchronizowanie pracy urządzeń, muszą być uwzględnione w algorytmach.



Rys. 5. Przykład algorytmu sterowania urządzeniem

Na strukturę i funkcje algorytmów najwyższego poziomu zasadniczy wpływ ma założony stopień swobody w adaptowaniu operatywnego planu zadań do aktualnej sytuacji w obiekcie oraz warunki wynikające z cech obiektu i cech procesu produkcyjnego. Inne będą algorytmy, gdy obiekt jest taśmą z wąskospecjalizowanymi urządzeniami i ustaloną wielkoseryjną produkcją. Natomiast będą inne, bardziej złożone, gdy obiekt składa się z jednostek o znacznym zakresie uniwersalności, z produkcją małoseryjną i wieloasortymentową. Wówczas na ogół, wymaga się (od algorytmów) bieżącego tzn. uwzględniającego aktualny stan obiektu - wyznaczania zadań dla jednostek lub zespołów jednostek. Przy tym wymaga się też optymalizacji o określonych kryteriach.

Algorytmy opracowywane na tym etapie nie mogą i nie powinny narzucać jakimi środkami w SKS będą one realizowane. Dobór środków, ich konfiguracji itp. będzie dokonany w następnych etapach.

Przyjętą - i chyba najbardziej przejrzystą - formą opisu algorytmów są schematy czynnościowe, ewentualnie sieci Petri. Uzupełnienie ich odpowiednimi komentarzami, słownikami, wyjaśnieniami symboli oznaczać itp. niewątpliwie ułatwi ich jednoznaczne odczytywanie. Często celowe jest także opracowanie uproszczonych flowdiagramów, z mniejszym stopniem szczegółowości, gdyż pełne ze względu na dużą liczbę szczegółów są mało przejrzyste. Zastąpienie pewnych fragmentów flowdiagramów tablicami decyzyjnymi znacznie je skracają.

Ponieważ opisy algorytmów są jednym z podstawowych dokumentów założeń na SKS, muszą one być jednoznaczne i dostatecznie precyzyjne.

Omówione wyżej zagadnienia są tylko nieznaczną częścią zagadnień, z którymi spotyka się projektant przy opracowywaniu algorytmów. Przy tym nawet dla wielu omówionych zagadnień nie podano spo-

sobów rozwiązania, gdyż brak jest ogólnych metod opracowywania algorytmów. Niewątpliwie pewnym ułatwieniem w tych pracach mogą być przykłady z już opracowanych algorytmów dla podobnych obiektów. Ponadto przy opracowywaniu algorytmów należy określić stany obiektu i stany SKS, przy których może nastąpić start i restart sterowania, a także możliwości i warunki przejęcia sterowania przez obsługę (lokalne ręczne sterowanie przez operatorów, zdalne ręczne sterowanie przez dyspozytora).

Funkcje sprzętu i oprogramowania w SKS

W opracowanych algorytmach (zadaniach sterowania) należy, przynajmniej wstępnie, dokonać podziału na fragmenty, które będą realizowane przez specjalizowany sprzęt i fragmenty realizowane programowo. Podział ten w zasadzie dotyczy algorytmów pierwszego poziomu. Wyższe poziomy z reguły są realizowane programowo.

Upowszechnienie się układów wielkiej skali integracji, w konfrontacji z kosztami i złożonością oprogramowania, podważa (słusznie dawniej) dążenie do programowej realizacji większości funkcji SKS, np. zliczanie, odmierzanie czasu, przekształcanie kodów, dekodowanie określonych stanów itp. obecnie w wielu wypadkach lepiej jest dokonywać za pomocą specjalizowanych układów.

Oczywiście jest, że funkcje, których algorytmy na danym etapie nie są jeszcze dostatecznie spracyzowane, należy pozostawić do realizacji programowej. Przy podziale należy też uwzględnić, że przy realizacji programowej łatwiej jest wyeliminować (odfiltrować) zakłócenia, łatwiejszy jest "start i restart" sterowania.

Dokonanie właściwego (optymalnego) podziału jest trudne i wymaga dobrze wyważonego kompromisu. Należy tutaj zaznaczyć, że modułowość sprzętu i elastyczność oprogramowania w znacznym stopniu łagodzą te trudności. Niemniej często lepiej jest ponownie rozważyć podział, gdy w następnym etapie będzie ustalona struktura sprzętu. Może okazać się, że w zestawie komputerowym (w oprogramowaniu) są duże niewykorzystane rezerwy mocy obliczeniowej.

Struktura sprzętu i struktura oprogramowania

W poprzednim etapie dokonano wstępnego podziału funkcji (zadań) SKS na realizowane sprzętowo i realizowane programowo. Na tym etapie powinno się więc ustalić właściwą strukturę sprzętu. Strukturę tę z jednej strony określa zestaw (mini-mikro) komputerów i powiązania (zależności) między nimi, z drugiej - układy zapewniające przesyłanie sygnałów między tym zestawem i urządzeniami obiektu (sprzężenie z obiektem). Do układów sprzężenia zalicza się transmisję sygnałów, układy komutacji, układy buforowego (separującego) pamiętania sygnałów itp. W odpowiednim też miejscu struktury powinny znaleźć się układy sprzętowe realizacji funkcji.

Przy doborze transmisji należy uwzględnić ilość przekaźowanych sygnałów, odległość punktów nadawania i odbioru oraz inne warunki pracy np. zakłócenia. Szczególnie przy większych ilościach sygnałów przesyłanych na większe odległości (ponad kilkadziesiąt m) konieczny jest właściwy dobór rodzaju transmisji (torów transmisyjnych) i sposobu ich wykorzystania.

Biorąc pod uwagę konieczną dla danego SKS "moc obliczeniową" (dla jego zadań realizowanych programowo) dobieramy zestaw komputerów. Może to być zestaw jednokomputerowy (struktura skupiona) lub wielokomputerowy (struktura rozłożona). W drugim wypadku m.in. należy ustalić sposób przekazywania danych między komputerami.

Właściwy dobór sprzętu często jest utrudniony przez:

- brak możliwości oszacowania z dostateczną dokładnością (na tym etapie) zapotrzebowania na "moc obliczeniową" (szybkość operacji, pojemność pamięci),
- ograniczony zbiór dostępnych komputerów i innego sprzętu standardowego,
- konieczność ustalenia przewidywanego zakresu rozszerzenia funkcji SKS wynikających z rozbudowy obiektu.

Innymi istotnymi czynnikami wpływającymi na strukturę sprzętu są:

- przewidywany zakres możliwości ingerencji obsługi w sterowanie obiektem i
- wymagania dotyczące ciągłości (niezawodności) sterowania.

Pierwszy czynnik pociąga konieczność wprowadzenia do struktury układów (urządzeń) umożliwiających komunikowanie się (konwersację) obsługi z SKS i urządzeń umożliwiających "obserwację" stanu procesu produkcyjnego na tablicach synoptycznych. Informacja zobrazowywana na tych tablicach może być dostarczana bezpośrednio z obiektu, bądź też za pośrednictwem układów (urządzeń) SKS odpowiednio przetworzona. Pierwszy sposób umożliwia "obserwację" procesu nawet przy zakłóceniach pracy pozostałych urządzeń SKS, natomiast drugi pozwala na dostarczanie informacji syntetycznej,

Drugi czynnik w skrajnym przypadku wymaga zdublowania sprzętu, tzw. "gorąca rezerwa". Jest to dla większości sytuacji ekonomicznie nieuzasadnione. W tym wypadku konieczna jest analiza skutków jakie pociągnie za sobą uszkodzenie (zakłócenie pracy) SKS któregoś z jego urządzeń. Tam, gdzie urządzenia obiektu wyposażone są w odpowiednie blokady zabezpieczające przed uszkodzeniami wynikającymi z zaniku lub niewłaściwego wystawiania, wystarczy powiadomienie obsługi o sytuacji awaryjnej. Możliwość ręcznego (zdalnego) sterowania urządzeniami obiektu m.in. zapewni sterowanie procesem w sytuacji awarii SKS i w czasie usuwania przyczyny tej awarii. Tablica synoptyczna z odpowiednią sygnalizacją o awarii i tablica ręcznego (zdalnego) sterowania zastępuje w pewnym zakresie "gorącą rezerwę".

W wyniku tych ustaleń można opracować schemat blokowy struktury sprzętu SKS, z którego powinno wynikać z jakich zespołów i układów składa się, jakie są połączenia między nimi i jak jest sprzężony obiekt z SKS (transmisja sygnałów). Dla tych zespołów i układów (z wyjątkiem zestawu komputerowego) powinny być opracowane uproszczone schematy logiczne (struktury rejestrowo-bramkowe) - jako rozwinięcie schematu blokowego. One powinny określać jak dany zespół (układ) przyjmuje, przetwarza i przekazuje dane z innych zespołów SKS.

Na ogół schemat blokowy i uproszczone schematy logiczne pozwalają prześledzić w jaki sposób sygnały o stanie i zdarzeniach w obiekcie docierają do komputera - jako podstawowej jednostki przetwarzającej oraz jak sygnały sterujące z komputera docierają do obiektu. Ponadto ze schematów powinien wynikać sposób adresowania przez programy przyjmowanych i wysyłanych danych.

Struktura oprogramowania ma zasadniczy wpływ nie tylko na efektywne wykorzystanie mocy obliczeniowej komputera, lecz także na pewne cechy eksploatacyjne SKS. Należą do nich między innymi:

- zakres możliwości ingerencji obsługi-dyspozytora w proces sterowania,
- szybkość i dokładność lokalizacji zakłóceń procesu sterowania,
- możliwość obserwowania przez obsługę procesu sterowania,
- zakres "odporności" oprogramowania na zakłócenia,
- zakres sprawdzania przez oprogramowanie wiarygodności otrzymywanych z obiektu danych,
- szybkość doprowadzania SKS (oprogramowania) do stanu "startowego i restartowego,
- możliwość odłączania i dołączania do sterowania z SKS poszczególnych urządzeń obiektu w trakcie procesu sterowania.

Dlatego właściwy dla danej struktury sprzętu i zadań SKS - dobór struktury oprogramowania ma tak istotne znaczenie. Struktura oprogramowania jest w zasadzie określoną strukturą oprogramowania podstawowego (system operacyjny + ogólne programy użytkowe dla danej klasy).

Na strukturę oprogramowania podstawowego mają wpływ następujące czynniki (poza samym sprzętem komputerowym):

- SKS jest w zasadzie systemem stałoprogramowym,
- wymagania na czas reakcji SKS na zdarzenia w obiekcie,
- sposób zbierania danych o stanie obiektu -
 - sporadyczne (zdarzenia są sygnalizowane przerwaniami)
 - cykliczne (wyznaczane przez zegar)
- zasada obsługi (przez program) zdarzeń
 - kolejność, priorytety,

- czy obsługa jednego zdarzenia może być przerwana przez inne zdarzenie,
- czy czas oczekiwania (przez program) na realizację przez urządzenie zlozonych czynności może być wykorzystany dla obsługi innych zdarzeń,
- czy w pamięci jest tworzony i aktualizowany obraz stanu obiektu oraz
- czy przewiduje się jego weryfikowanie z rzeczywistym stanem obiektu.

Innym istotnym zagadnieniem jest opracowanie właściwych opisów oprogramowania. Jest to konieczne tym bardziej, że najczęściej eksploatacją oprogramowania, wnoszeniem zmian i uzupełnień zajmują się inne osoby, nie autorzy. Często konieczne, szczególnie przy uruchamianiu, jest prześledzenie - rozkaz po rozkazie - jak program zareagował na dane zdarzenie w obiekcie. W innym wypadku konieczne jest ustalenie dlaczego - przy jakim wewnętrznym stanie oprogramowania - nastąpiło zakłócenie procesu sterowania. Dlatego też opis oprogramowania, oprócz klasycznych tabulogramów, powinien zawierać schematy czynnościowe (flowdiagramy) i schematy blokowe o różnym stopniu szczegółowości, w odpowiedni sposób powiązane z tabulogramami. Przy tym język opisów powinien być zbliżony do języka użytkownika SKS.

Wnioski

Konieczne jest prowadzenie prac, w wyniku których powinny być wypracowane ogólne - przynajmniej dla danej klasy obiektów - metody umożliwiające z dostateczną dokładnością:

- dokonywanie analizy i opisu obiektu z punktu widzenia projektanta SKS,
- określanie zadań SKS,
- opracowywanie i opisywanie algorytmów sterowania i procesu w obiekcie,
- dokonywanie oszacowań koniecznej "mocy obliczeniowej" zestawów komputerowych SKS,
- opisywanie oprogramowania (i sprzętu) SKS.

Pozwoli to wyeliminować część trudności, z jakimi spotyka się w pierwszych etapach projektant SKS, a tym samym przyspieszyć prace i zmniejszyć ryzyko błędów. Możliwe jest opracowanie pewnych wzorców, szablonów projektowania SKS, które po weryfikacji w praktyce stałyby się obowiązujące. Jednocześnie mogłoby to przyczynić się do pewnej unifikacji pojęć-określeń, ułatwiając współpracę różnych zespołów zaangażowanych w projektowaniu SKS.

Literatura

- [1] HARRISON T. -ed.: Handbook of Industrial Control Computers. T. 1 i 2. Moskwa: MIR 1975
- [2] MARTIN J.: Programowanie maszyn cyfrowych w systemach uwarunkowanych czasowo. Warszawa: WNT 1970
- [3] NIEDERLIŃSKI A.: Systemy cyfrowe automatyki przemysłowej. T. 1 i 2. Warszawa: WNT 1977.

Cz.2. System komputerowego sterowania centrum SKSC-1

Wstęp

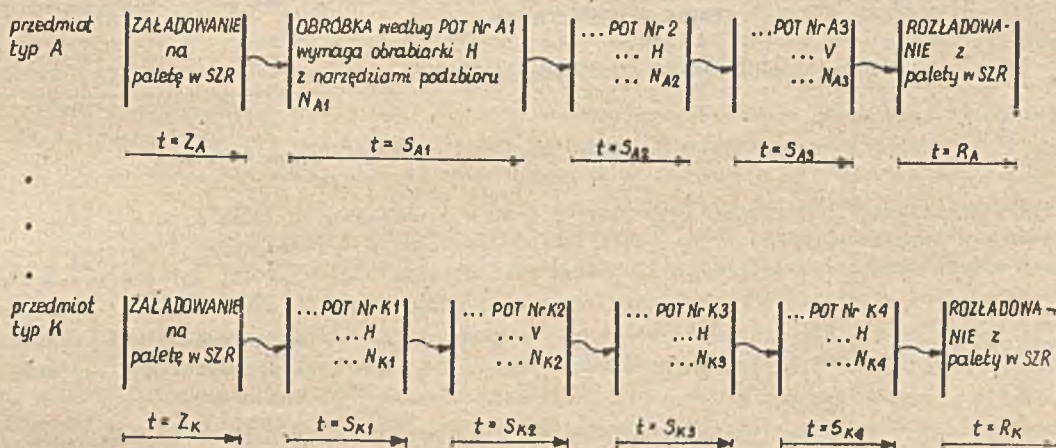
Zautomatyzowany wydział obróbki skrawaniem - jakim jest centrum produkcyjne CP-KOR-1 - może być dobrym przykładem obiektów często występujących w przemyśle maszynowym. Jednocześnie, na podstawie wielu jego cech, może być uznany za obiekt należący do klasy omawianej w I części artykułu.

W tej części artykułu, w znacznym uproszczeniu, omówimy podstawowe cechy CP i SKSC-1.

Centrum produkcyjne - obiektem sterowanym przez SKSC-1

Przeznaczenie CP

Zadaniem CP jest obróbka skrawaniem przedmiotów - odlewów korpusowych wymagających wykonania licznych skomplikowanych operacji (wiercenia, frezowania itp.). W CP jednocześnie mogą być obrabiane partie różnych typów przedmiotów. Każdy typ przedmiotu ma ustaloną kolejność programów operacji technologicznych (POT), według których ma on być obrabiany. Z kolei obróbka według danego POT dokonana może być w odpowiedniej i odpowiednio wyposażonej w narzędzia obrabiarkie. Czasy obróbki są różne dla różnych POT.



Rys. 1. Proces obróbki poszczególnych typów przedmiotów

Proces produkcyjny rozpoczyna się od zamocowania (załadowania) "surowego" przedmiotu na specjalną paletę, na której będzie on dostarczany kolejno do ustalonych dla danego typu przedmiotu obrabiarek i tam poddany obróbce.

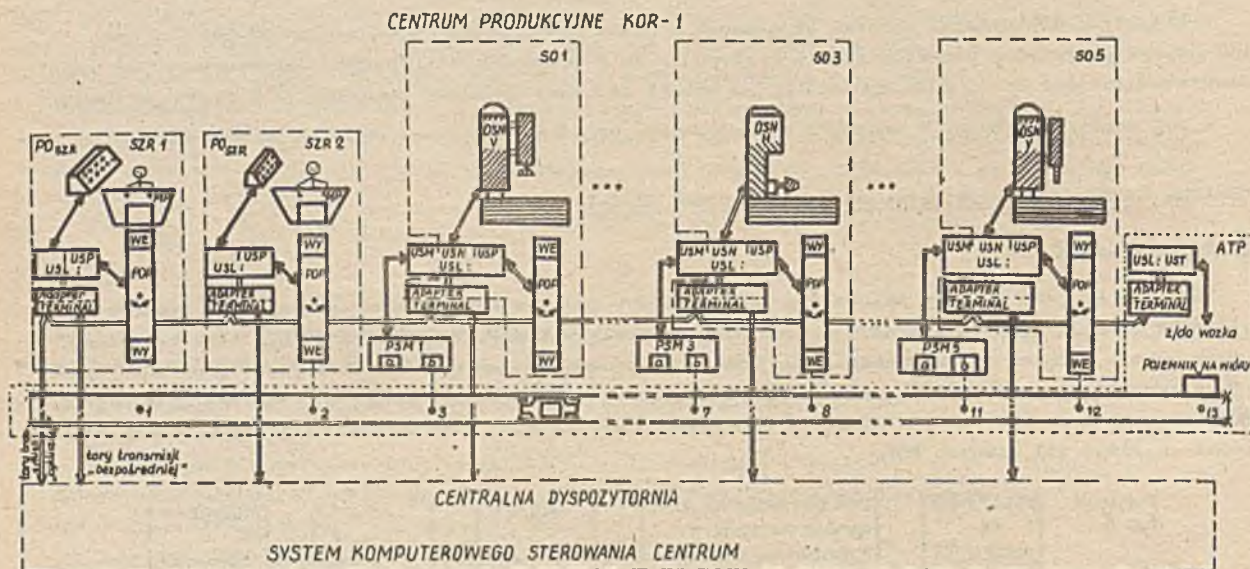
Po zakończeniu obróbki we wszystkich ustalonych obrabiarkach, przedmiot, jako globalnie obrabiany, jest rozładowywany z palety. Na tym kończy się proces produkcyjny danego przedmiotu. Poza załadunkiem i rozładunkiem dokonywanym przez operatora cały proces odbywa się bez udziału człowieka.

Średni czas obróbki przedmiotu w jednej obrabiarence wynosi 20 min, średnio przedmiot obrabiany jest w trzech obrabiarkach. Stąd produkcja w CP jest w zasadzie produkcją jednostkową różnych typów przedmiotów.

Produkcja CP nie jest bezpośrednio powiązana z produkcją innych obiektów zakładu - na wejściu i wyjściu są odpowiednio duże magazyny surowych i globalnie obrabianych przedmiotów. Konieczne zewnętrzno powiązania uwzględnia się w planowaniu zadań dla CP.

Jednostki funkcjonalne i struktura CP

Dla realizacji wyżej wymienionego zadania CP wyposażono jest w niżej wymienione środki produkcji - rys. 2.



Rys. 2. Schemat strukturalny jednostek funkcjonalnych CP-KOR-1

• Obrabiarki sterowane numerycznie OSN (cztery typu H - poziomego i jedna typu V - pionowego) wyposażone w automatyczne magazyny narzędzi. Zbiory narzędzi w tych magazynach są dobierane do bieżących zadań. Obrabiarki są przystosowane do automatycznego przyjęcia i obróbki przedmiotów zamocowanych na paletach. Po wprowadzeniu (przyjęciu) przedmiotu na stół obrabiarki konieczne jest dostarczenie - z biblioteki POT SKSC-1 - do układu sterowania numerycznego USN ciągu znaków odpowiedniego POT. Do identyfikacji wprowadzanej palety (i pośrednio przedmiotu) służy czujnik kodu palety.

Z obrabiarką jest związany podajnik operacyjny palet POP - uzupełniony układem sterowania lokalnego USP (układ sterowania podajnikiem), który odpowiednio wystawiony z SKSC-1 zabiera lub oddaje paletę z/do obrabiarki lub z/na wózek transportowy. Ramiona POP są wyposażone w czujniki kodu palety.

Wymieniony zestaw urządzeń i układów stanowi jednostkę funkcjonalną nazywaną stacją obrabiarową - SO. Poza przygotowaniem do startu (na początku zmiany) i ewentualnie wymianą narzędzi w magazynie, jednostka nie wymaga obsługi przez człowieka.

• Stacja załadunkowo-rozładunkowa SZR (dwie) jest jednostką funkcjonalną, w której operator w miejscu przeobrażenia - MP ładuje (mocuje) na palecie przedmiot przeznaczony do obróbki w CP, lub rozładunkowo go po obróbce w CP. Jakiego typu i kiedy przedmiot ma być załadowany określa SKSC-1 wydając operatorowi (za pośrednictwem pulpitu POsZR) odpowiednie polecenie. Podobnie, kiedy i który globalnie obrabiany przedmiot ma być dostarczony do SZR dla rozładunku - określa SKSC-1. MP wyposażony jest w czujnik kodu palety. Oprócz wymienionego MP z pulpitem POsZR, w zestaw jednostki wchodzi podajnik POP, którego rola jest identyczna jak w SO. Poza czynnościami bezpośrednio związanymi z ładowaniem i rozładunkiem przedmiotu jednostka podobnie jak SO, nie wymaga obsługi przez człowieka.

o Pomocnicza stacja magazynowa - PSM (pięć) wyposażona jest w dwa zasobniki (a i b), w każdym może być przechowana paleta z przedmiotem (lub bez przedmiotu). Stacja wyposażona jest w układ sterowania lokalnego USM. Odpowiednioysterowanu (z SKSC-1) stacja pobiera, lub oddaje paletę z/na wózek transportowy do/z zasobnika. Każdy zasobnik wyposażony jest w czujnik kodu palety.

Wymienione urządzenia i układ stanowią jednostkę funkcjonalną (PSM), która podobnie jak SO nie wymaga obsługi przez człowieka.

o Automatyczny transport palet - ATP (jeden) składa się z wózka transportowego WT poruszającego się po torowisku, wzdłuż którego rozmieszczone są wyżej wymienione stacje. ATP wyposażony jest w układ sterowania lokalnego UST. Odpowiednio sterując (z SKSC-1), można skierować wózek do dowolnej stacji. Tam POP lub zasobnik PSM może oddać paletę na wózek lub zabrać z wózka. Wózek wyposażony jest w pojemnik wiórów, do którego w trakcie przewożenia są usuwane wióry z palety i przedmiotu. Po napełnieniu się pojemnika (sygnał do SKSC-1) wózek powinien być skierowany przez SKSC-1 do stacji 13, gdzie pojemnik jest opróżniony. Wózek wyposażony jest w czujnik kodu palety.

Wymienione urządzenia i układy stanowią jednostkę funkcjonalną ATP, która podobnie jak SO nie wymaga obsługi przez człowieka.

o Palety PL (20 szt.). Konstrukcja palety umożliwia zamocowanie na niej za pomocą specjalnych uchwytów przedmiotu z wymaganą dokładnością (bazowanie), przenoszenie jej przez POP i wózek ATP, oraz jej mocowanie na stole obrabiarki. W każdej paletce znajdują się kołki kodowe ustalające jej numer, który może być zidentyfikowany przez czujniki kodu. Identyfikacja przedmiotu możliwa jest tylko pośrednio przez identyfikację numeru palety. W zasadzie palety nie "opuszczają" CP i mogą "przebywać" tylko w urządzeniach z czujnikami kodu - umożliwia to SKSC-1 śledzenie ich "przepływu".

Układy sterowania lokalnego USL (USN, USP, UST, USM) wyposażone są w odpowiednie pulpity umożliwiające obsłudze lokalne ręczne sterowanie danymi urządzeniami. Pozwala to na wykonanie określonych czynności przy przygotowywaniu urządzenia do sterowania z SKSC-1 oraz ręczne sterowanie urządzeniem w szczególnych sytuacjach (zakłócenia procesu, awarie).

W hali CP, w odpowiednich miejscach (szafach) umieszczono terminale specjalnej transmisji sygnałów z/i do SKSC-1. Tam też umieszczono układy adapterów dopasowujące pod względem elektrycznym sygnały z USL do standardu przyjętego w SKSC-1 (ściślej w transmisji). Przyjętym standardem jest TTL. Ustalono, że linia podziału obiekt CP-SKSC-1 przebiega przez "styk" adapter-terminal. Pisząc dalej o sygnałach przekazywanych z CP(USL) do SKSC-1 i odwrotnie, rozumiemy się przekazywanie z adaptera do terminalu i odwrotnie. Układy sterowania lokalnego uznano za integralną część obiektu (urządzeń). Dlatego charakterystyki-cechy jednostek (urządzeń) będą rozpatrywane łącznie z USL.

o Struktura CP. Stacje obrabiarkowe, stacje załadowczo-rozładowcze i pomocnicze stacje magazynowe są samodzielnymi urządzeniami, mogącymi w zasadzie pracować wzajemnie niezależnie (asynchronicznie). Jednak w trakcie procesu produkcyjnego do SO i SZR powinny być dostarczane, odpowiednie i w odpowiednim czasie, palety z przedmiotami. Realizuje to odpowiednioysterowany transport ATP.

ATP jest transportem "scentralizowanym" umożliwiającym pobranie palety z dowolnej stacji i przewiezienie jej do innej dowolnej.

o Blockady i zabezpieczenia. Urządzenia CP i ich układy sterowania są wyposażone w odpowiednie blockady i zabezpieczenia przed niewłaściwymysterowaniem

Ogólne zadania SKSC-1

Zadania SKSC-1 podzielono na cztery zasadnicze grupy.

- o Sterowanie pracą stacji obrabiarkowych SO, to jest:
 - dostarczanie - na ządania z USN-ów poszczególnych SO - znaków odpowiedniego POT, przy tym szybkość dostarczania nie powinna zmniejszać rytmu pracy OSN;
 - odpowiednieysterowanie POP, gdy zakończy się obróbka przedmiotu w OSN lub do SO przybył wózek transportowy.
- o Sterowanie przepływem w CP palet z przedmiotami:

- na podstawie analizy stanu (zgłoszeń) SI i SZR określenie, która z jednostek ma być obsługiwana przez transport, tzn. skąd i dokąd ma być przewieziona paleta;
- na podstawie powyższego - odpowiednie wystrojenie wózka transportowego;
- do tej grupy zaliczono też sterowanie PSM.

- Sterowanie wprowadzaniem do CP nowych przedmiotów do obróbki, tj.:
 - określanie kiedy i jaki przedmiot ma być wprowadzany do CP (załadowany w SZR),
 - określenie kiedy i jaką paletę z obrobionym przedmiotem wyprowadzić z CP (rozładować w SZR),
 - odpowiednio do powyższych wystrojenia SZR.
- Przyjmowanie z systemu nadrzędnego planu zadań operatywnych dla CP oraz sporządzanie odpowiednich raportów o przebiegu procesu.

Obrabiarki OSN są podstawowymi urządzeniami produkcyjnymi CP, gdyż w nich odbywa się obróbka przedmiotów. Sama obróbka przedmiotów w poszczególnych OSN przebiega wzajemnie niezależnie. Jednak dla maksymalnego wykorzystania czasu pracy OSN konieczne jest możliwie szybkie dostarczenie przedmiotu do obróbki, gdy tylko zakończy się obróbka poprzedniego. Konieczny przedmiot powinien być już w tym czasie dostępny, tzn. obrobiony w poprzednich SO. Podobnie obrobiony przedmiot powinien być jak najszybciej zabrany z OSN, aby można było wprowadzić tam kolejny przedmiot do obróbki. Wymaganie to znajduje swoje odbicie zarówno w zadaniach SKSC-1 grupy drugiej i trzeciej, jak też w planowaniu zadań operatywnych dla CP.

Gdy z jakiegoś powodu konieczne jest wyłączenie lub ręczne sterowanie któregoś, lub kilku urządzeń (uszkodzenie konserwacja) pozostałe nadal mogą być sterowane przez SKSC-1. Przewidziano także możliwość - w określonym zakresie - ingerencji obsługi (dyspozytora) w proces sterowania.

Przebieg sterowania procesem produkcyjnym

Gdy w SO ma być obrabiany przedmiot, dla którego dana SO jest pierwsza w ustalonym ciągu (por. rys. 1) powstaje zapotrzebowanie na wprowadzenie tego typu przedmiotu do CP. SKSC-1 analizując przebiegający w CP proces produkcyjny, prognozuje chwile, w których pojawi się to zapotrzebowanie. Z odpowiednim wyprzedzeniem w czasie, wybierając wolne SZR, zleca operatorowi rozpoczęcie ładowania przedmiotu na paletę. Po załadowaniu operator zgłasza (do SKSC-1) wykonanie polecenia.

Wysyłając odpowiednie polecenia do wózka ATP, SKSC-1 powoduje przewożenie palety z przedmiotem do ustalonych SO, jeśli jest tam wolne miejsce (na ramieniu WE podajnika POP), albo do PSM, gdzie przedmiot będzie oczekiwał na zwolnienie się tego miejsca.

W kolejnych OSN przedmiot jest poddawany obróbce zgodnie z dostarczonym przez SKSC-1 programem POT. Gdy w OSN zakończy się obróbka przedmiotu, sterowany przez SKSC-1 podajnik POP zabierze go i umieści tam nowy. Dla nowego przedmiotu, SKSC-1 odszuka w bibliotece odpowiedni POT i będzie go znakami dostarczać (na ządanie) do USN. Obrobiony w danej SO przedmiot powinien być odwieziony do określonej następnej SO lub do SZR.

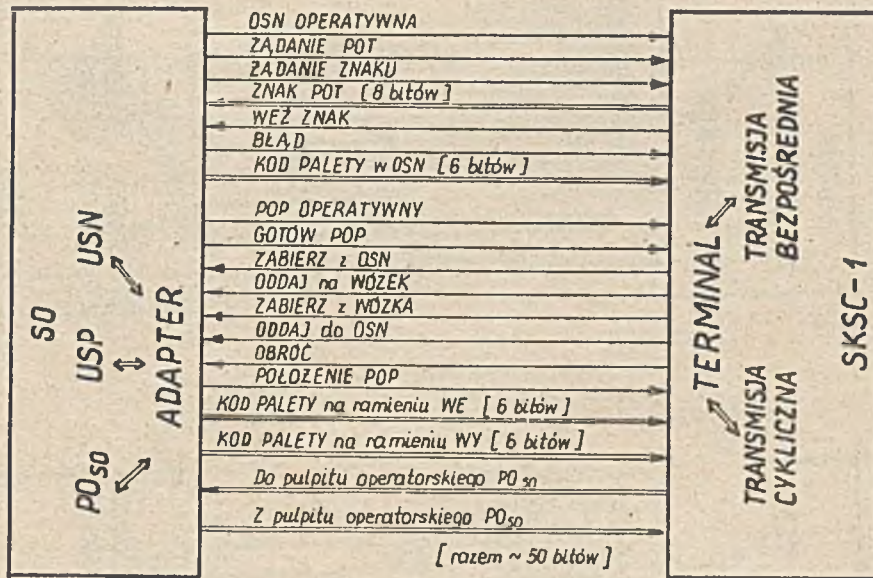
Po dostarczeniu do SZR - ściślej do MP - palety z obrobionym przedmiotem, SKSC-1 wydaje operatorowi polecenie rozładowania. Gdy operator zawiadomi o wykonaniu polecenia, SKSC-1 może wydać polecenie załadowania nowego przedmiotu na zwolnioną paletę.

Ważniejsze zdarzenia procesu są rejestrowane przez SKSC-1 i mogą być wyprowadzane w formie raportu z przebiegu procesu.

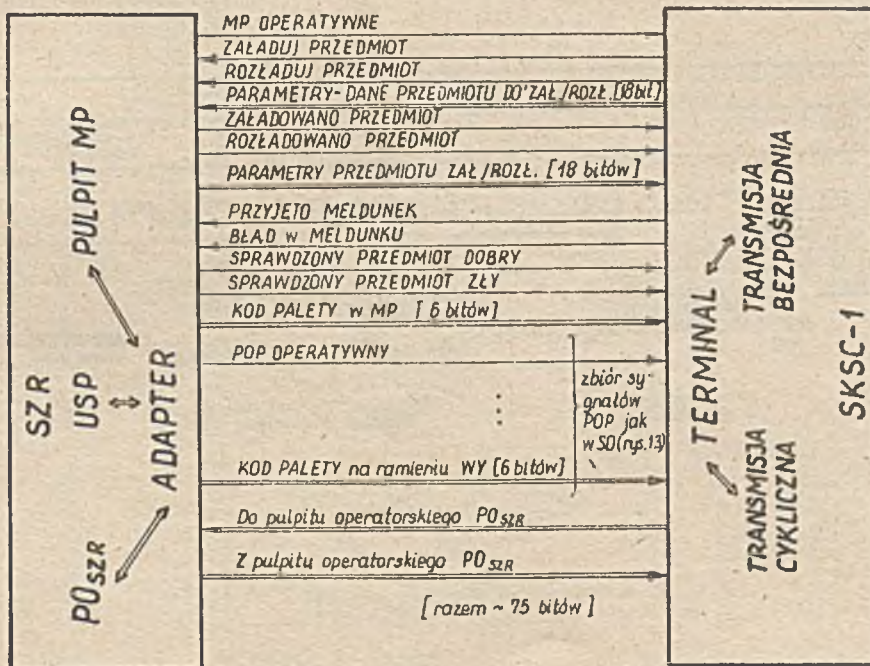
Zasady współpracy jednostek CP z SKSC-1

Dla bardziej szczegółowego zorientowania czytelnika z zakresem "samosterowności" i "podatności" na zewnętrzne sterowanie jednostek CP, niżej podano zasady ich współpracy z SKSC-1.

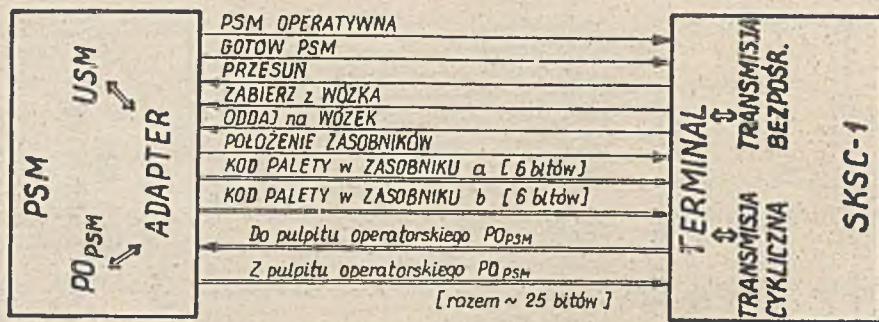
SO Zbiór sygnałów, harmonogramy związane ze sterowaniem SO oraz uproszczone algorytmy pracy OSN i POP podane są na rys. 3, 7, 8, 9, 10. Gdy OSN jest przygotowana do pracy przy sterowaniu z SKSC-1 (to znaczy jest sprawna i przełączona na sterowanie z komputera), do SKSC-1 posyłany jest sygnał OSN OPERATYWNA.



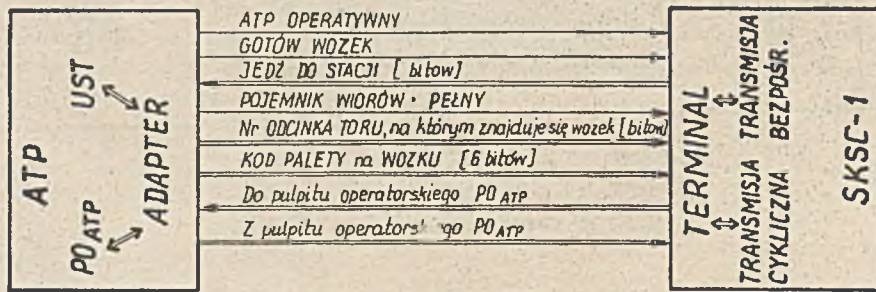
Rys. 3. Zbiór sygnałów SO ↔ SKSC-1



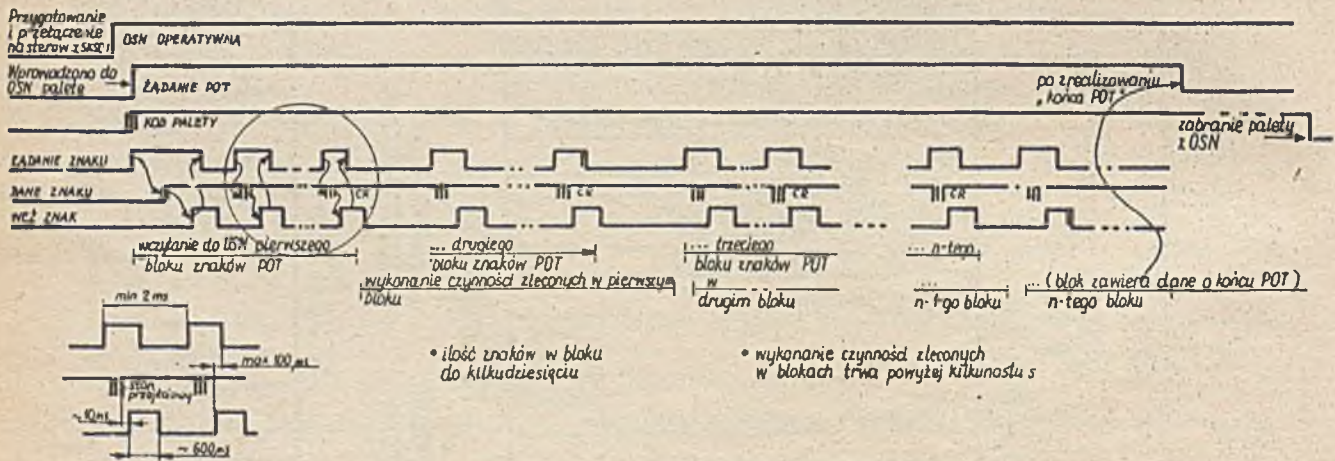
Rys. 4. Zbiór sygnałów SZR ↔ SKSC-1



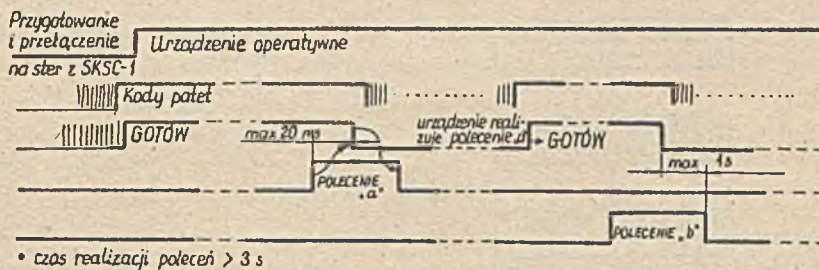
Rys. 5. Zbiór sygnałów PSM SKSC-1



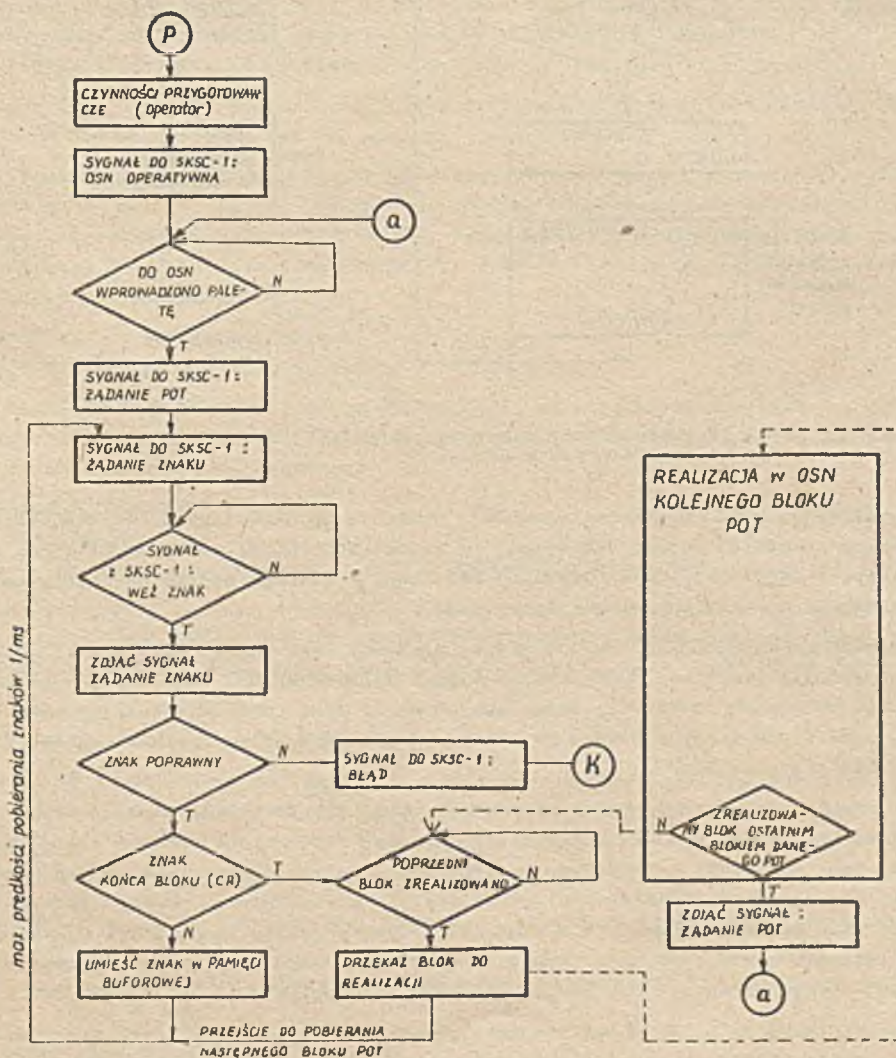
Rys. 6. Zbiór sygnałów ATP SKSC-1



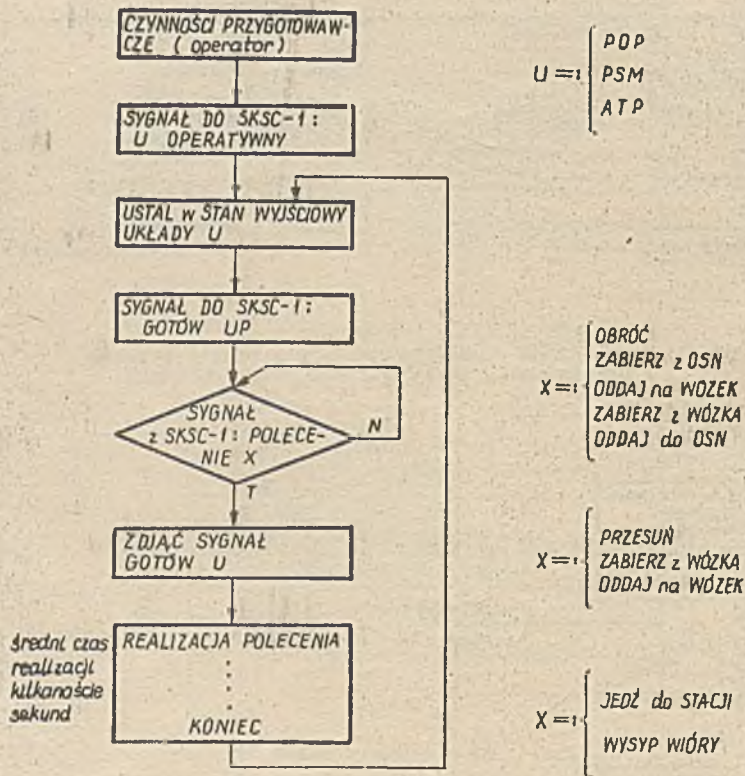
Rys. 7. Harmonogram przesyłania POT



Rys. 8. Harmonogram przekazywania i realizacji poleceń w urządzeniach POT(USP); ATP(UST); PSM(USM)



Rys. 9. Uproszczony algorytm pracy OSN (współpraca z SKSC-1)



Rys. 10. Uproszczony algorytm pracy POP ATP, PSM
współpraca z SKSC-1

Wprowadzenie palety na stół obrabiarki spowoduje wysłanie do SKSC-1 sygnałów ŻĄDANIE POT i ŻĄDANIE ZNAKU. Nieco wcześniej pojawi się sygnał (6 bitów) KOD PALETY w OSN określający numer wprowadzonej palety. W SKSC-1 zostanie określony POT, według którego będzie obrabiany przedmiot (numer tego POT będzie wyświetlony na pulpicie POso).

Gdy SKSC-1 umieści na szynach ZNAK POT (8 bitów) pierwszy znak danego POT, wysyła do USN sygnał WEŹ ZNAK. Po pobraniu przez USN znaku zanika sygnał ŻĄDANIE ZNAKU. Jego kolejne pojawienie się oznacza żądanie kolejnego znaku POT. Odpowiadając na to SKSC umieszcza kolejny znak i wysyła sygnał WEŹ ZNAK. Jeżeli USN wykryje błąd w przekazywanych znakach POT, zamiast sygnału ŻĄDANIE ZNAKU posyła do SKSC-1 sygnał BŁĄD.

W ten sposób przebiega pobieranie przez USN kolejnych znaków POT. Ciąg znaków całego POT jest podzielony na bloki (kilkanaście do kilkudziesięciu znaków) rozdzielone znakiem specjalnym CR. Po brany blok znaków jest pamiętany w buforowej pamięci USN i po wykonaniu przez OSN czynności zleconych poprzednim blokiem jest przekazywany do realizacji. Do zwolnionej pamięci buforowej jest pobierany kolejny blok na POT. Informacja o końcu POT zawarta jest w samym POT i wykrywana jest przez USN. O zakończeniu w OSN ostatnich czynności zleconych danym POT (koniec obróbki i ustawienie stołu w pozycji wymiany palety) USL powiadamia SKSC-1 zdjęciem sygnału ŻĄDANIE POT.

Urządzeniem manipulacyjnym w SO, umożliwiającym przekazywanie palety z wózka transportu do OSN i odwrotnie, jest podajnik POP. Ramiona POP mogą też być wykorzystywane jako "bufory" łagodzące, w pewnym zakresie, wymagania na czas dostarczania i zabierania palety z SO. Na przykład, następna paleta z przedmiotem do obróbki w SO powinna być dostarczona, gdy tylko ramię wejściowe POP jest wolne, bez oczekiwania na zakończenie w OSN obróbki poprzedniego przedmiotu.

Gdy POP jest przygotowany do pracy przy sterowaniu z SKSC-1 (to znaczy jest sprawny i przełączony na sterowanie z komputera) do SKSC-1 posyłany jest sygnał POP OPERATYWNY.

Sygnałom oznaczającym, że POP może przyjąć polecenie z SKSC-1 jest sygnał GOTÓW POP. POP. wraz z USP, przyjmuje z SKSC-1 i realizuje następujące polecenie:

- ZABIERZ z WÓZKA - spowoduje zabranie palety z wózka;
- ODDAJ do OSN - spowoduje oddanie palety do OSN;
- ZABIERZ z OSN - spowoduje zabranie palety z OSN;
- ODDAJ na WÓZEK - spowoduje oddanie palety na wózek;

Polecenia to będą zrealizowane tylko przy odpowiednim ustawieniu ramion POP (np. przy zabieraniu z wózka, ramię WE powinno być ustawione od strony transportu).

- OBRÓĆ - spowoduje zmianę ustawienia ramion POP (obrót o 180°).

O położeniu-ustawieniu ramion POP, SKSC-1 jest informowany sygnałem POŁOŻENIE POP.

Przyjęcie polecenia powoduje zanik sygnału GOTÓW POP, a to z kolei powinno spowodować zanik sygnału-polecenia. Po zrealizowaniu polecenia ponownie pojawia się sygnał GOTÓW POP. Informacja o numerze palety znajdującej się na danym ramieniu jest przekazywana do SKSC-1 odpowiednimi sygnałami: KOD PALETY na ramieniu WE (6 bitów), KOD PALETY na ramieniu WY (6 bitów).

SO wyposażona jest w pulpit operatorski POso - zawierający zestaw przełączników, przycisków, wyświetlaczy - umożliwiający obsłudze komunikowanie się z SKSC-1.

SZR Zbiór sygnałów, harmonogram związany ze sterowaniem SZR oraz uproszczone algorytmy pracy MP i POP podane są na rys. 4, 8, 10. Gdy MP jest sprawny i operator może przyjmować polecenia z SKSC-1 z SZR jest wysyłany sygnał MP OPERATYWNE. Informacja o numerze palety znajdującej się w MP jest przekazywana do SKSC-1 sygnałom KOD PALETY w MP (6 bitów). Operator przyjmuje i realizuje następujące polecenia przesyłane z SKSC-1:

- ROZŁADUJ PRZEDMIOT z palety znajdującej się w MP;
- ZAŁADUJ PRZEDMIOT na paletę znajdującą się w MP.

Jednocześnie sygnałem PARAMETRY PRZEDMIOTU DO ZAŁ/ROZŁ, są przekazywane z SKSC-1 dane o typie przedmiotu; przyjęcie polecenia operator potwierdza odpowiednim przyciskiem na pulpicie powodując wysłanie do SKSC-1 jednego z sygnałów:

- ROZŁADOWYWANIE PRZEDMIOTU
- ŁADOWANIE PRZEDMIOTU

a to powoduje w SKSC-1 zdjęcie sygnału PARAMETRY PRZEDMIOTU DO ZAŁ/ROZŁ.

Po wykonaniu polecenia operator wyciskając przycisk na pulpicie powoduje, że zanika odpowiedni sygnał ROZŁADOWYWANIE PRZEDMIOTU lub ŁADOWANIE PRZEDMIOTU. Wcześniej operator na odpowiednich nastawnikach pulpitu wstawia dane przedmiotu, a to powoduje pojawienie się (wysyłanie do SKSC-1) sygnału:

- PARAMETRY PRZEDMIOTU ZAŁadowanego/ROZładowanego

Gdy ten meldunek o wykonaniu jest zgodny z poleceniem, SKSC-1 zdejmuje odpowiedni sygnał-polecenie ROZŁADUJ PRZEDMIOT lub ZAŁADUJ PRZEDMIOT. W razie niezgodności - SKSC-1 posyła sygnał

- BŁĄD W MELDUNKU,

który zaniknie, razem z odpowiednim sygnałem, gdy SKSC-1 otrzyma poprawny meldunek.

Za pośrednictwem pulpitu w SZR może być przekazana do SKSC-1 informacja o obróbkach przedmiotów:

- DOBRY Dla przesyłania danych o przedmiocie wykorzystuje się sygnał DANE PRZEDMIOTU ZAŁ/ROZŁ.
- ZŁY

Przyjęcie meldunku SKSC-1 potwierdza sygnałem:

- PRZYJĘTO MELDUNEK

POP w SZR spełnia te same funkcje i jest tak samo sterowany jak POP w SO.

PSM Zbiór sygnałów, harmonogram związany ze sterowaniem PSM oraz uproszczony algorytm pracy PSM podane są na rys. 5, 8, 10.

Gdy PSM jest przygotowana do pracy przy sterowaniu z SKSC-1, tzn. jest sprawna i przelączona na sterowanie, z komputera do SKSC-1 posyłany jest sygnał PSM OPERATYWNA.

Sygnałem, że PSM może przyjąć z SKSC-1 polecenie, jest sygnał GOTÓW PSM. PSM wraz z USM przyjmuje z SKSC-1 i realizuje następujące polecenia:

- ZABIERZ z WÓZKA - spowoduje zabranie palety z wózka do zasobnika znajdującego się w pozycji wymiany,
- ODDAJ na WÓZEK - spowoduje oddanie palety na wózek z zasobnika znajdującego się w pozycji wymiany,
- PRZESUŃ - spowoduje zmianę zasobnika w pozycji wymiany ("a" na "b" lub odwrotnie "b" na "a").

Informacja o tym, który zasobnik znajduje się w pozycji wymiany jest przekazywana do SKSC-1 sygnałem POŁOŻENIE ZASOBNIKÓW.

Przyjęcie polecenia powoduje zanik sygnału GOTÓW PSM, a to z kolei powinno spowodować zanik sygnału polecenia. Po zrealizowaniu polecenia ponownie pojawia się sygnał GOTÓW PSM.

Dane o numerze palety znajdującej się w zasobniku są przesyłane do SKSC-1 sygnałami: KOD PALETY w ZASOBNIKU a, KOD PALETY w ZASOBNIKU b.

ATP Zbiór sygnałów, harmonogram związany ze sterowaniem ATP oraz uproszczony algorytm pracy ATP podane są na rys. 6, 8, 10.

Gdy ATP jest przygotowany do pracy przy sterowaniu z SKSC-1, do SKSC-1 jest wysyłany sygnał ATP OPERATYWNY. Sygnałem oznaczającym, że ATP może przyjąć polecenie z SKSC-1, jest sygnał GOTÓW WÓZEK. ATP wraz z UST przyjmuje z SKSC-1 i realizuje następujące polecenia:

- JEDŹ DO STACJI - spowoduje udanie się wózka do wskazanej stacji,
- WYSYP WIÓRY - spowoduje wysypanie wiórów z pojemnika.

Numer odcinka (stacji), na którym znajduje się wózek jest przekazywany do SKSC-1 sygnałem Nr ODCINKA.

Przyjęcie polecenia przez ATP powoduje zanik sygnału GOTÓW WÓZEK, a to z kolei powinno spowodować zanik sygnału-polecenia. Po zrealizowaniu polecenia, ponownie pojawia się sygnał GOTÓW WÓZEK.

Napełnienie się pojemnika wiórów powoduje wysłanie do SKSC-1 sygnału POJEMNIK PEŁNY.

Dane o numerze palety znajdującej się na wózku są przekazywane sygnałem KOD PALETY na WÓZKU.

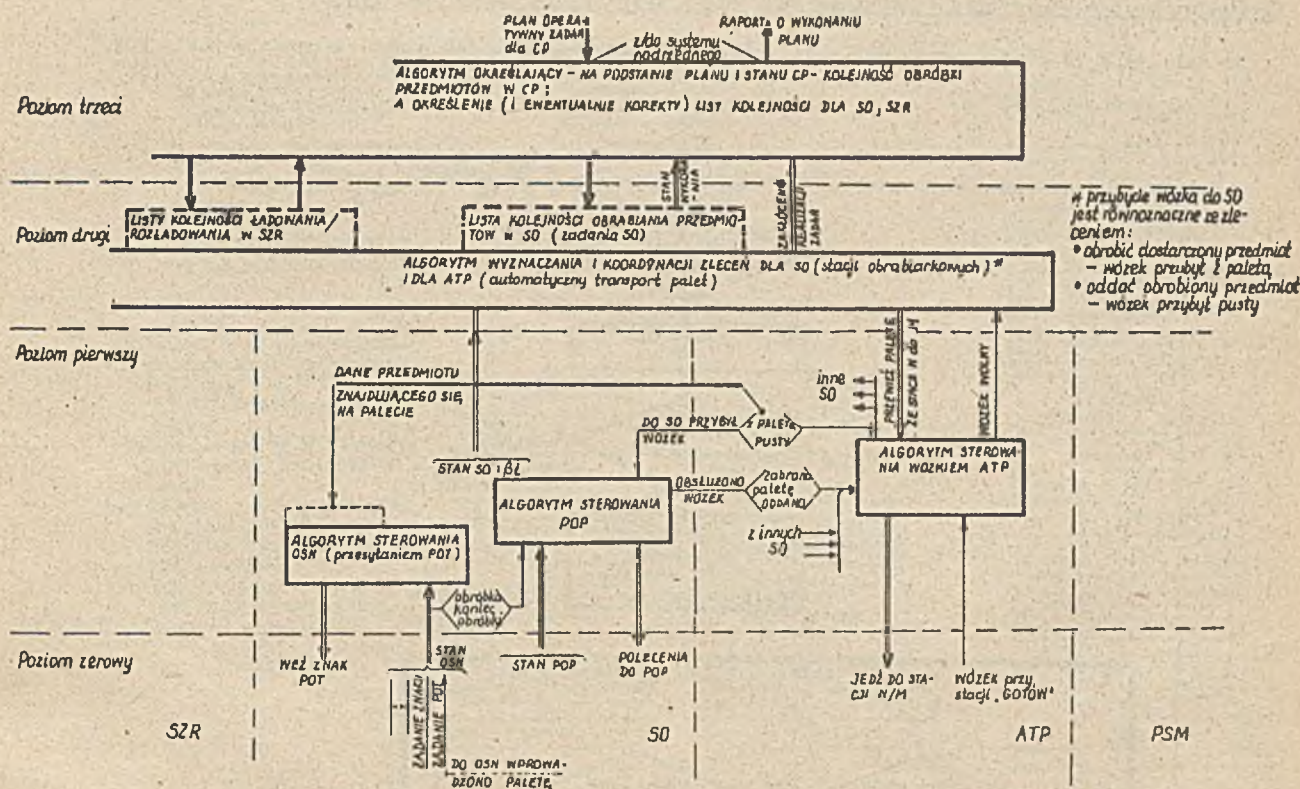
Wyżej podane opisy zasad współpracy są uproszczone, uwzględniają tylko sekwencje stanów jednostki przy normalnej (niezakłóconej) jej pracy. Ze względu na ramy artykułu pominięto stany szczególne, w których może znaleźć się jednostka wskutek zakłóceń lub ingerencji obsługi. Niemniej zagadnienia te - na etapie poznawania obiektu - były szczegółowo opracowane i wpływają one w znacznym stopniu na zadania SKSC-1.

Algorytmy sterowania CP przez SKSC-1

Trzy poziomy algorytmów

W wyniku analizy opisu CP i opisu zadań SKSC-1 zawartych w poprzednim punkcie przyjęto trzypięciową strukturę algorytmów sterowania (rys. 11).

Algorytmy poziomu pierwszego związane są z bezpośrednim sterowaniem urządzeniami, jak OSN i POP w stacjach obrabiarkowych, wózkiem ATP oraz (nie pokazanymi na rysunku) magazynami PSM i stacjami SZR. Każdy z tych algorytmów otrzymując zlecenia z drugiego poziomu posyła odpowiednie i w odpowiedniej chwili polecenia do urządzenia - "steruje" urządzeniem. Wykonanie zleceń jest sygnalizowane do poziomu drugiego przez stan urządzenia.



Rys. 11. Struktura powiązań algorytmów sterowania CP

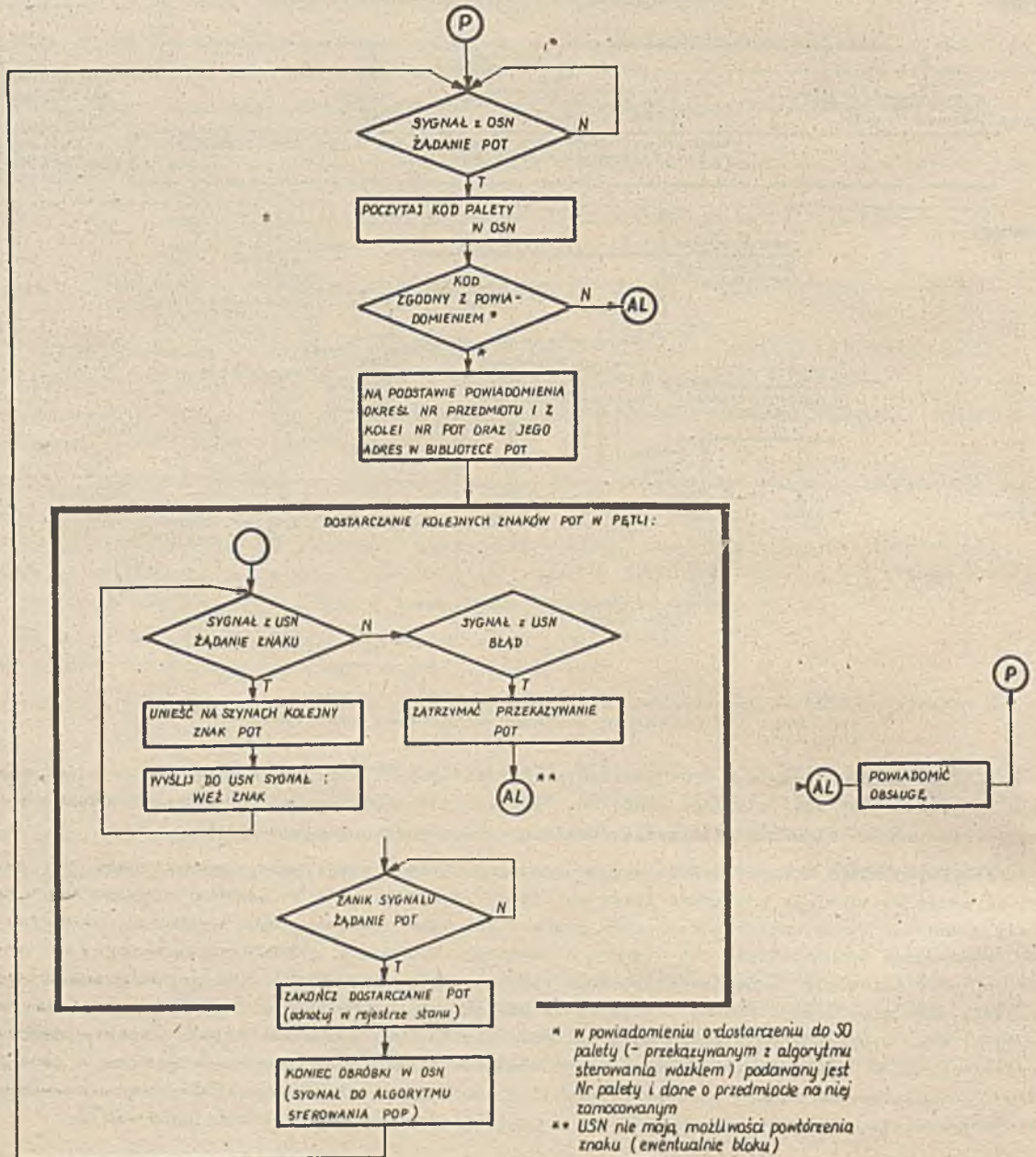
Algorytmy poziomu drugiego - na podstawie list kolejności obrabiania przedmiotów w poszczególnych SO (oraz listy SZR) i stanów urządzeń - wyznaczają odpowiednie i posyłają w odpowiedniej chwili zlecenia do algorytmu sterowania wózkiem - koordynują pracę urządzeń.

Algorytmy poziomu trzeciego powinny, uwzględniając plan operacyjny i zaawansowanie procesu w CP, określać ewentualnie korygować listy kolejności dla SO i SZR. To powinno zapewnić koordynację pracy jednostek funkcjonalnych w trakcie procesu tak, aby spełnione było kryterium optymalizacji, tzn. maksymalne wykorzystanie czasu pracy obrabiarek. Układanie optymalnych harmonogramów pracy OSN-ów, przy założeniu jednoczesnej obróbki różnych typów przedmiotów jest zagadnieniem trudnym. Stąd dla pierwszej wersji SKSC-1 przyjęto, że zagadnienie to będzie rozwiązywane w systemie nadrzędnym, tzn. dysponującym większą mocą obliczeniową i plan operacyjny będzie zawierał omawiane listy kolejności. Nie jest to rozwiązanie najwłaściwsze, gdyż zakłócenia występujące w jednej ze stacji obrabiarkowych wymaga znacznych korekt. Właściwym rozwiązaniem byłoby bieżące uzupełnianie (korygowanie) tych list przez SKSC-1 z uwzględnieniem aktualnego stanu jednostek*.

Zasady współpracy SKSC-1 i systemu nadrzędnego, przy ustalaniu i korygowaniu harmonogramów pracy CP są opisane w artykule: M. Kuopczak - Zagadnienia układania i realizacji harmonogramów pracy Centrum Produkcyjnego. Zeszyty Naukowe Politechniki Śląskiej 1978 nr 580.

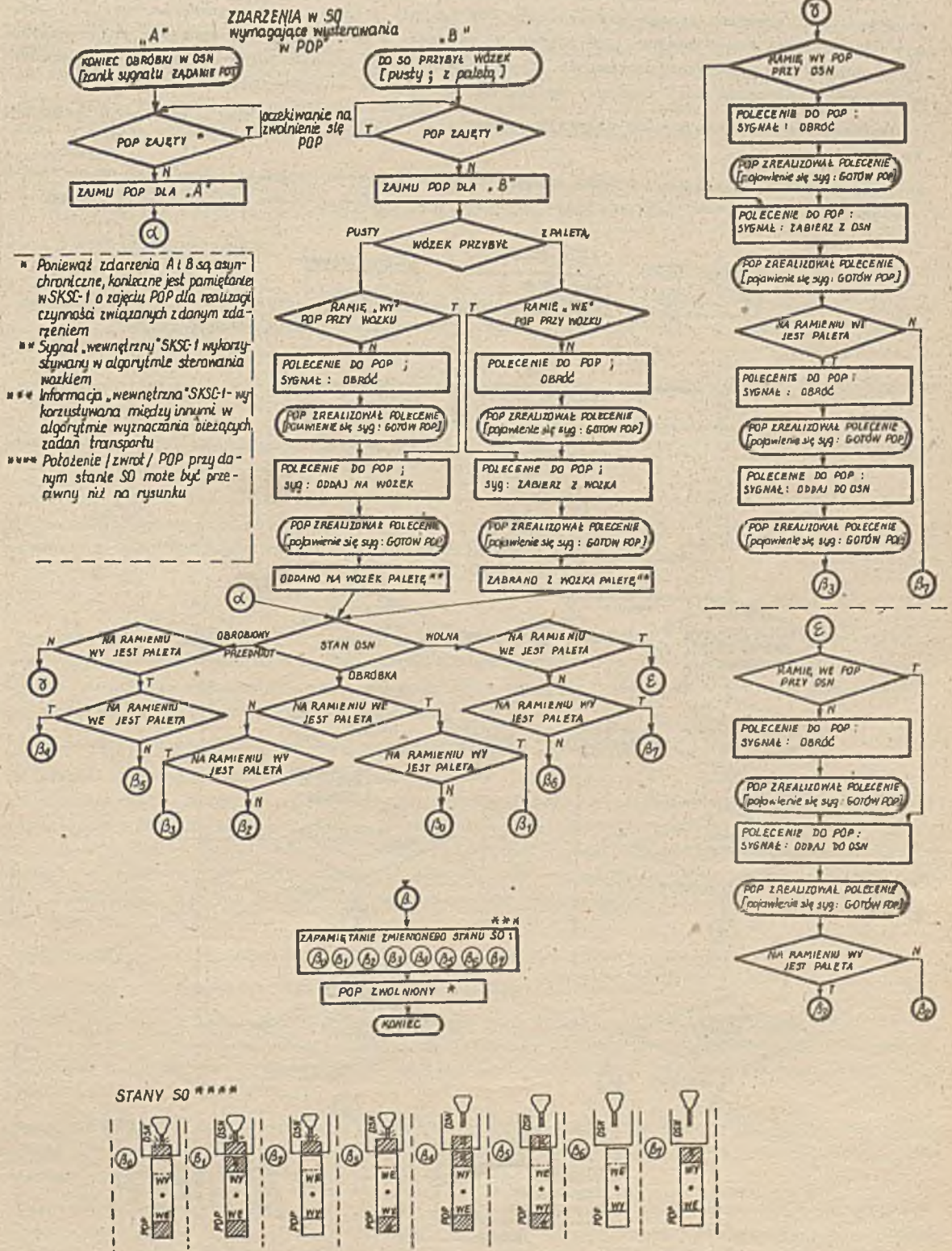
Uproszczony opis algorytmów sterowania OSN, POP i wyznaczania zadań dla wózka

Na rys. 12 i 13 podano przykłady uproszczonych flowdiagramów algorytmów sterowania OSN i POP.



Rys. 12. Uproszczony algorytm przesyłania znaków POT

Sterowanie OSN sprowadza się do dostarczania - na ządania z USN - znaków odpowiedniego POT (por. rys. 9). Dla określenia tego POT konieczna jest identyfikacja przedmiotu wprowadzonego do OSN. Dokonuje tego SKSC-1 pośrednio przez odczyt kodu palety w OSN. Z kolei, zostaje określone miejsce w "bibliotece POT" gdzie znajduje się dany POT, skąd jest on fragmentami pobierany i znakami przekazywany do USN.



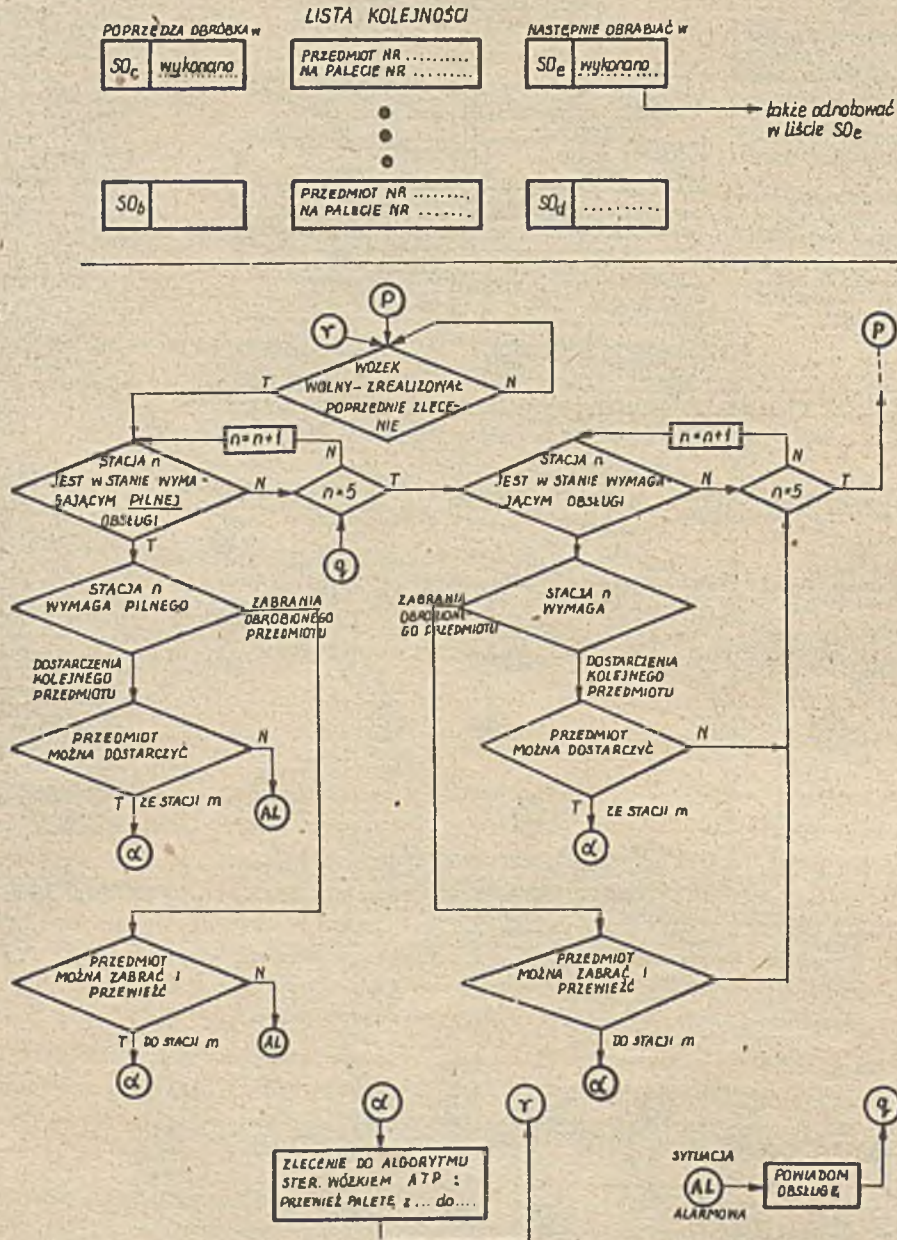
Rys. 13. Uproszczony algorytm sterowania POP
(w zależności od stanu SO)

O zakończeniu w OSN obróbki przedmiotu, SKSC-1 jest powiadamiany zdjęciem sygnału ŻADANIE POT.

Zakończenie obróbki wymaga wystawienia POP zgodnie z algorytmem podanym na rys. 13 - zdarzenie "A". Ogólnie rzecz biorąc, POP "dąży" do zabrania z OSN palety z obrobionym przedmiotem i wprowadzenia tam kolejnego przedmiotu. Aby można było to wykonać muszą być spełnione warunki: ramię WY musi być wolne i na ramieniu WE powinna być paleta z przedmiotem do obróbki.

Na tym samym rysunku podano algorytm czynności POP dla zdarzenia "B" - przybycie do SO wózka transportu.

Wystąpienie w SO sytuacji-stanów szczególnych (alarmowych) sygnalizowane jest przez SKSC-1 obsłudze. Na flowdiagramach podano tylko niektóre z wykrywanych przez SKSC-1 stanów szczególnych.



Rys. 14. Uproszczony algorytm wyznaczania zleceń dla wózka transportu

Na rys. 14 podano uproszczony flowdiagram algorytmu wyznaczania zleceń dla sterowania wózkiem. Jest to przykład koordynacji pracy danej SO z innymi jednostkami.

Gdy wózek zrealizuje poprzednie zlecenie sprawdza się, czy stan któregoś ze SO wymaga pilnej obsługi. Jeśli tak, to sprawdza się czy można wykonać tę obsługę. Jeśli nie - to sytuacja szczególna AL, jeśli tak - to odpowiednie zlecenie do algorytmu sterowania wózkiem.

Jeśli żadna z SO nie wymaga pilnej obsługi, to sprawdza się czy któraś ze stacji może być obsłużona w trybie obsługi zwykłej. Gdy tak - to wysyła się odpowiednie zlecenie do algorytmu sterowania wózkiem.

Wyżej, w bardzo skróconej formie opisano tylko przykłady trzech algorytmów. Z braku miejsca nie opisano bardzo wielu innych algorytmów realizowanych przez SKSC-1.

Sprzęt i oprogramowanie SKSC-1

Podział na podsystemy

Uwzględniając m.in. strukturę obiektu, charakter poszczególnych zadań wyznaczonych dla SKSC-1 oraz moc obliczeniową dostępnych w tym czasie (r. 1973) krajowych minikomputerów całość zadań SKSC-1 podzielono na trzy podsystemy:

- podsystem sterowania obrabiarkami PSO, do zadań którego należy dostarczanie z "biblioteki" do USN programów POT oraz sterowanie podajnikami POP w SO,
- podsystem sterowania transportem PST, do zadań którego należy koordynacja (na podstawie listy kolejności) pracy SO i SZR oraz sterowanie wózkiem ATP,
- podsystem sterowania dyspozycyjnego PSD, do zadań którego należy ustalanie i aktualizowanie (z uwzględnieniem otrzymywanego z systemu nadrzędnego planu zadań CP) list kolejności oraz sterowanie załadowywaniem i rozładowywaniem palet w SZR.

Sprzęt

Dla zapewnienia podatności na zmiany, większość funkcji związanych z zadaniami SKSC-1 jest realizowana programowo w standardowych zestawach minikomputerów. Układy specjalizowanego sprzętu w zasadzie zapewniają tylko odpowiednie przekazywanie sygnałów (komutacja i buforowe pamiętanie) między zestawami minikomputerowymi i urządzeniami CP.

Do realizacji zadań SKSC-1 dobrano pokazaną na rys. 15 strukturę sprzętu.

W pamięci dyskowej zestawu minikomputerowego ZM1 zorganizowana jest biblioteka programów POT oraz umieszczone są programy podsystemu PSO, którego "kopie" są ściągane w miarę potrzeb do pamięci operacyjnej JC1. Do kanału multipleksorowego JC1 dołączona jest specjalizowana jednostka JSM1 zawierająca układy 5 podkanałów związanych z dostarczaniem znaków POT do USN-ów. Do kanału programowanego (arytmometru) JC1 dołączona jest jednostka JSA1 składająca się ze zbioru rejestrów, bramek i układów generujących przerwania. Zapewniają one przekazywanie sygnałów między minikomputerem i wszystkimi SO.

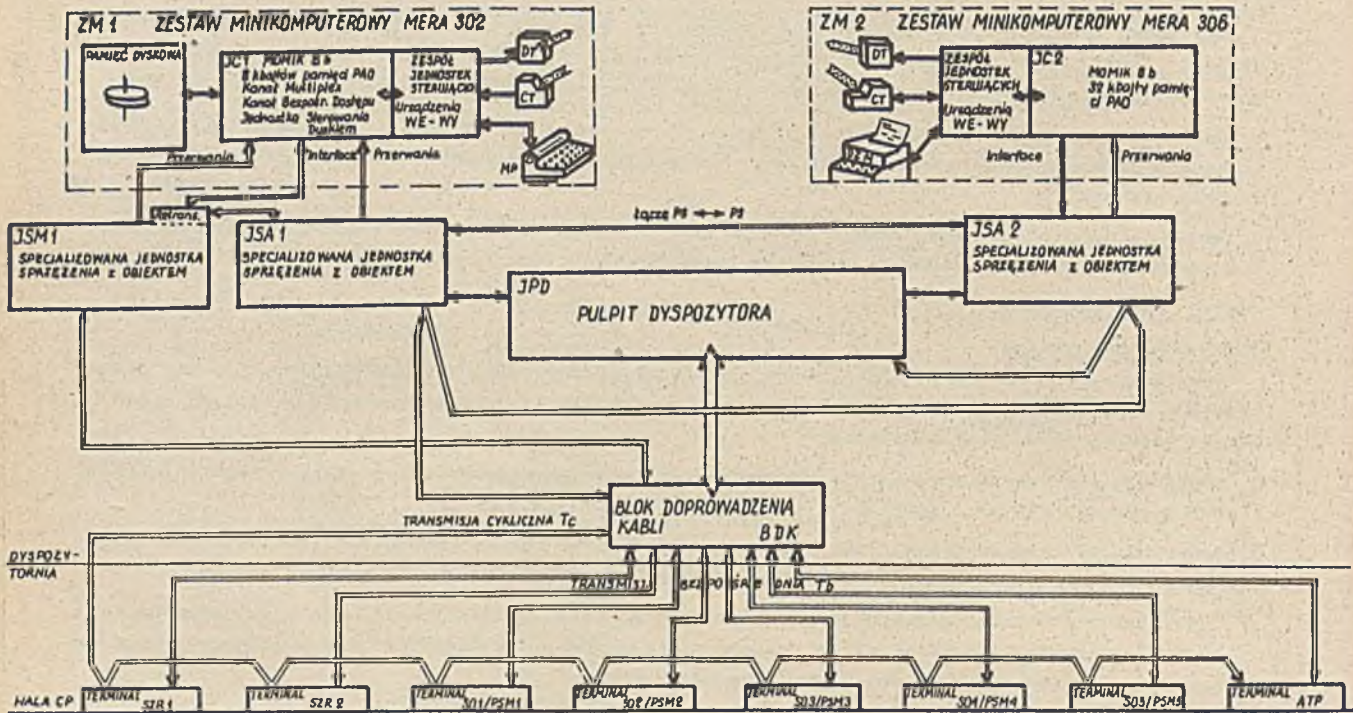
W jednostkach JSM1 i JSA1 znajdują się także układy transmisji sygnałów.

Zestaw ZM1 oraz jednostki JSM1 i JSA1 stanowią sprzęt podsystemu PSO.

Zestaw ZM2 wraz z dołączoną do niego jednostką JSA2 stanowią sprzęt pozostałych dwóch podsystemów PST i PSD. Oprogramowanie obu podsystemów rezyduje w pamięci operacyjnej JC2. JSA2 spełnia analogiczną rolę jak JSA1 i składa się z takich samych układów. Za jej pośrednictwem procesor JC2 śledzi i steruje SZR, ATP i PSM.

Ponieważ pomiędzy podsystemami PSO i PST/PSD istnieją powiązania funkcjonalne, konieczna jest wzajemna wymiana informacji między nimi. Wymianę tę (bajtami) zapewniają układy specjalnego łączącego podsystem - podsystem (PS-PS) umieszczone w JSA1 i JSA2.

Pulpit dyspozytora JPD umożliwia obsłudze śledzenie na bieżąco pracy poszczególnych urządzeń CP (nawet przy awarii któregoś z podsystemów). W szczególnych przypadkach dyspozytor może przejąć sterowanie urządzeniami, co w pewnym zakresie spełnia rolę "gorącej rezerwy". Dlatego pulpit wy-



Rys. 15. Schemat struktury sprzętu SKSC-1

posażony jest nie tylko w układy związane z wyświetlaniem informacji o stanie urządzeń CP, ale także w układy umożliwiające ręczne sterowanie.

Blok doprowadzenia kabli nie zawiera żadnych układów. Jest to tablica przyłączy kablowych wyposażona w odpowiednie złącza.

Oprogramowanie

Ograniczona pojemność pamięci operacyjnej, jednoczesność pracy OSN, sposób przyjmowania przez USN znaków POT oraz parametry pamięci dyskowej - wymagają odpowiedniej organizacji przechowywania i przesyłania POT.

Przy wprowadzaniu do biblioteki, POT jest dzielony za pomocą specjalnego programu na fragmenty mieszczące się w sektorze dysku (192 bajty) i zawierające całkowitą liczbę bloków. Koniec każdego fragmentu jest oznaczony znakiem specjalnym LT. Fragmenty te są zapisywane i przechowywane w kolejnych sektorach dysku. Adres początku danego POT (pierwszego sektora) jest zapisany w katalogu. Wprowadzanie POT do biblioteki odbywa się off-line.

Gdy do OSN zostanie wprowadzony przedmiot, w katalogu będzie odszukany adres pierwszego sektora odpowiedniego POT i sektor ten zostanie ściągnięty do określonego "bufora" w pamięci operacyjnej minikomputera. Następnie wysyłany jest sygnał startu do podkanału związanego z tą OSN; przysyłając zapytania, będzie mógł pobierać kolejno znaki POT. W pamięci operacyjnej znajdują się odrębny "bufor" dla każdego podkanału. Multiplexorowa praca podkanałów zapewnia możliwość "jednocześnie" przekazywania znaków POT do wszystkich USN.

Gdy w przokazywanych znakach wykryty będzie znak LT, do bufora danego podkanału zostanie ściąg-

nięty kolejny sektor*. Czas na "dojście" kolejnego sektora nie jest krytyczny, ponieważ do USN przekazany jest cały blok, a jego realizacja w OSN trwa stosunkowo długo.

W celu umożliwienia wykrywania przez SKSC-1 nieprawidłowej pracy określonych urządzeń CP, w pamięci SKSC-1 jest tworzony i aktualizowany przewidywany "obraz" stanu tych urządzeń. Gdy przy porównaniu tego obrazu z rzeczywistym, odczytanym stanem urządzeń stwierdza się niezgodność, może to oznaczać nieprawidłową pracę. Przewiduje się też wprowadzenie kontroli przekroczenia maksymalnego czasu wykonywania przez urządzenie czynności zleconych danym poleceniem.

Struktura SKSC-1 umożliwia sterowanie urządzeniem CP:

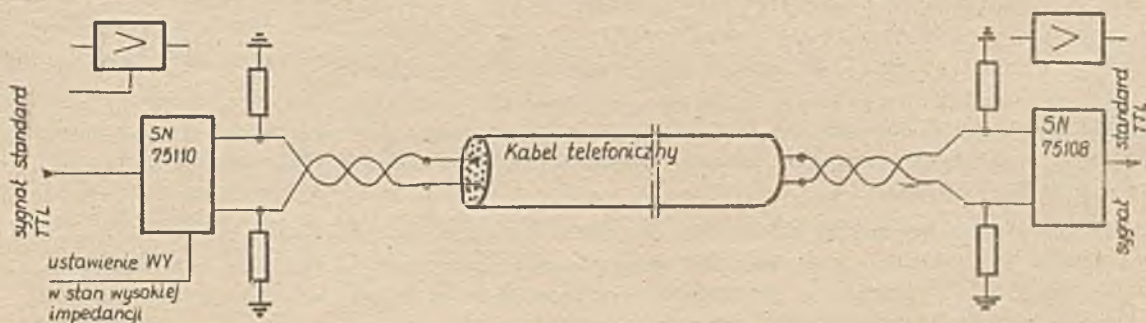
- ręczne z lokalnego pulpitu przy urządzeniu,
- ręczne zdalne z pulpitu dyspozytora,
- automatyczne komputerowe przez SKSC-1.

Przełączenie danego urządzenia na sterowanie lokalne, "odcina" możliwość sterowania z pulpitu dyspozytora i minikomputera. Z kolei sterowanie z pulpitu uniemożliwia sterowanie z minikomputera. Przełączenie to może nastąpić - awaryjnie - w dowolnym momencie. Jednak takie połączenie może wywołać zakłócenia w SKSC-1. Dlatego zalegany jest inny sposób. Obsługa zgłasza do SKSC-1 "prośbę" o zezwolenie na przejęcie sterowania danym urządzeniem, a SKSC-1 wydaje takie zezwolenie przy odpowiednim stanie urządzenia i programów w minikomputerze. Wyżej opisano tylko niektóre funkcje realizowane przez oprogramowanie.

Transmisja sygnałów między SKSC-1 i CP (terminalami)

Liczba sygnałów-bitów, które muszą być przekazywane między terminalami (w CP) i jednostkami w dyspozytorni (w SKSC-1) wynosi ponad 500. Przy odległościach między nadajnikami i odbiornikami sygnałów dochodzących do kilkudziesięciu metrów i przy znacznym polu zakłóceń wymagało to opracowania odpowiedniej transmisji sygnałów.

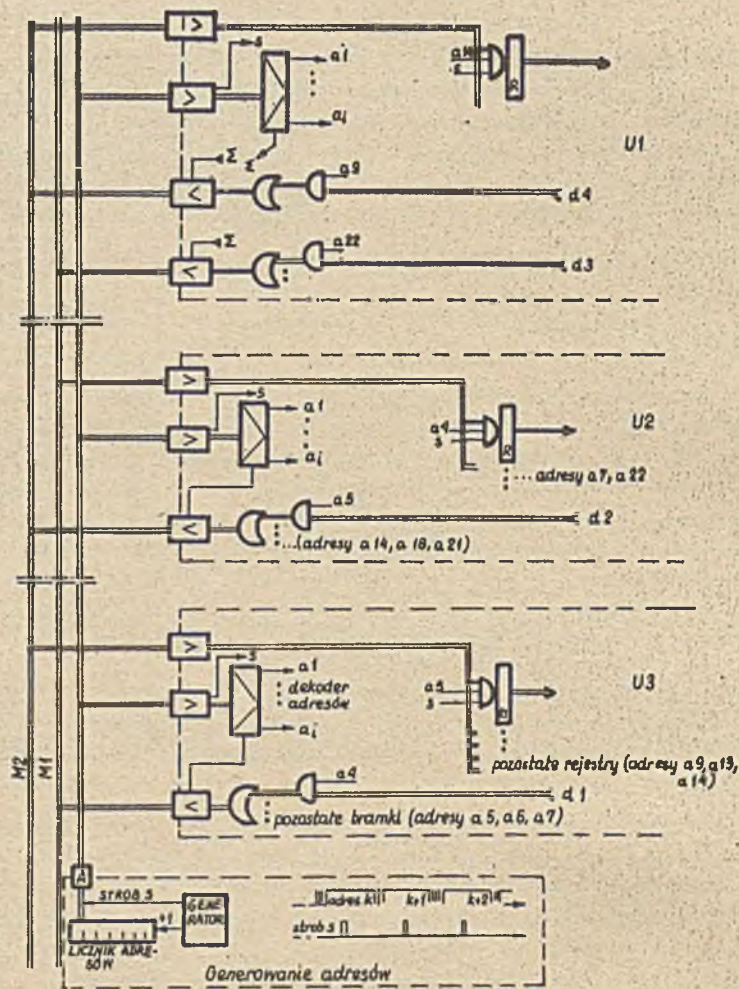
Po badaniach odporności na zakłócenia wybrano do realizacji tor transmisji pokazany na rys. 16. Ustawienie wyjścia nadajnika w stan wysokiej impedancji umożliwia "równoległe" dołączanie nadajników do danego toru przesyłania. Za wyborem tej wersji przemawiały jej prostota i względy ekonomiczne. Niewątpliwą wadą toru jest brak izolacji między urządzeniami w hali a urządzeniami w dyspozytorni (jak np. przy zastosowaniu optoizolatorów).



Rys. 16. Schemat toru transmisji sygnału

W celu zmniejszenia liczby torów przesyłania zorganizowano przesyłanie części informacji wspólnymi torami "w podziale czasu" - transmisja cykliczna. To. Zasada pracy transmisji cyklicznej wynika ze schematu rys. 17.

* Istnieje też możliwość wprowadzania określonych "wstawek" między kolejne sektory. Związane to jest z potrzebą korygowania i uzupełniania POT w trakcie jego przesyłania.



Rys. 17. Zasada pracy transmisji cyklicznej

W układzie generowania adresów wytwarzany jest cyklicznie ciąg 128 adresów oraz impuls strobujący. Strob - usytuowany w "środku" adresu - zapewnia odczytywanie lub wpisywanie do rejestru informacji w stanie ustalonym.

W przedziale czasu wyznaczonym adresem a4, na szyny magistrali M1 jest przekazywana informacja d1 (z układu U3). Informacja ta może być odczytywana lub zapisana do rejestru w dowolnym z układów, np. na rys. 17 w układzie U2. Podobnie informacja d2 z układu U2, przekazywana magistralą M2 w czasie trwania adresu a14 jest odczytywana w U1 i w U3.

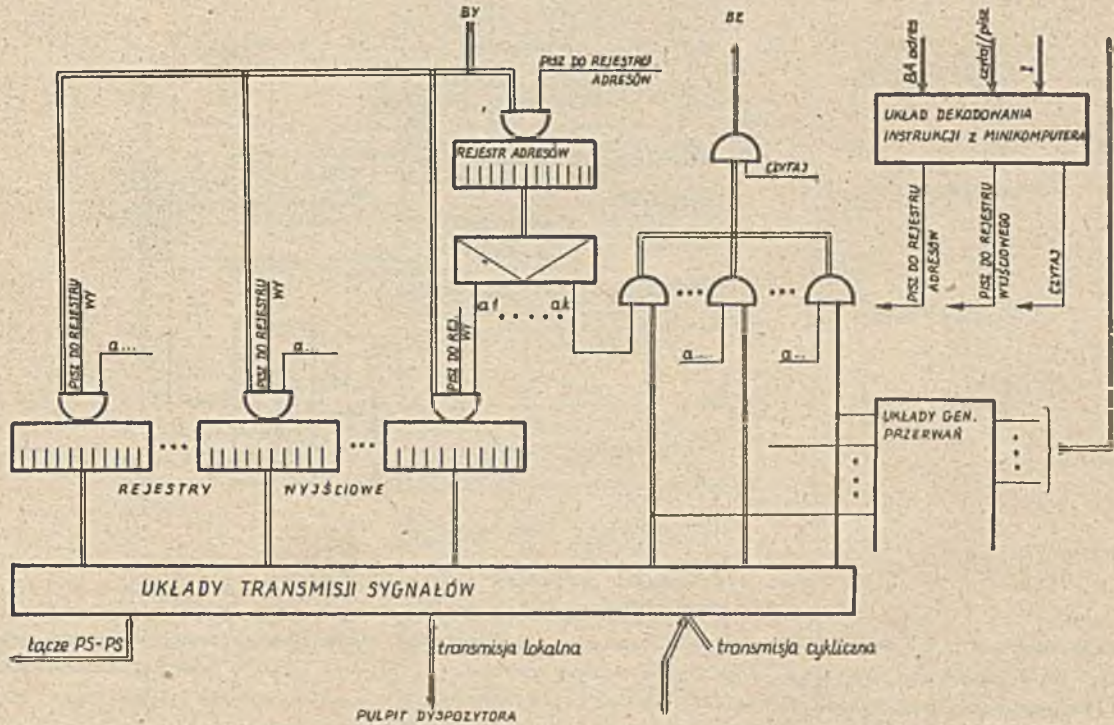
Transmisją cykliczną jest przekazywanych ~ 350 bitów. Pozostała część informacji jest przekazywana transmisją bezpośrednią, gdzie każdy bit informacji ma osobny tor przesyłania.

Przesyłanie danych między jednostkami wewnątrz dyspozytorni (transmisja lokalna) odbywa się torami, z zastosowaniem jako nadajnika układu SN 7407, przy tym każdy bit ma wydzielony tor.

Specjalizowane jednostki sterujące

Zasady przesyłania i przyjmowania informacji przez minikomputer (określone instrukcją wejścia/wyjścia i interfejsem) oraz ilość i forma informacji, która ma być przekazana między CP i SKSC-1,

narzuciły określoną strukturę jednostek specjalizowanych JSA1 i JSA2. Ogólną strukturę jednostki pokazuje rys. 18.



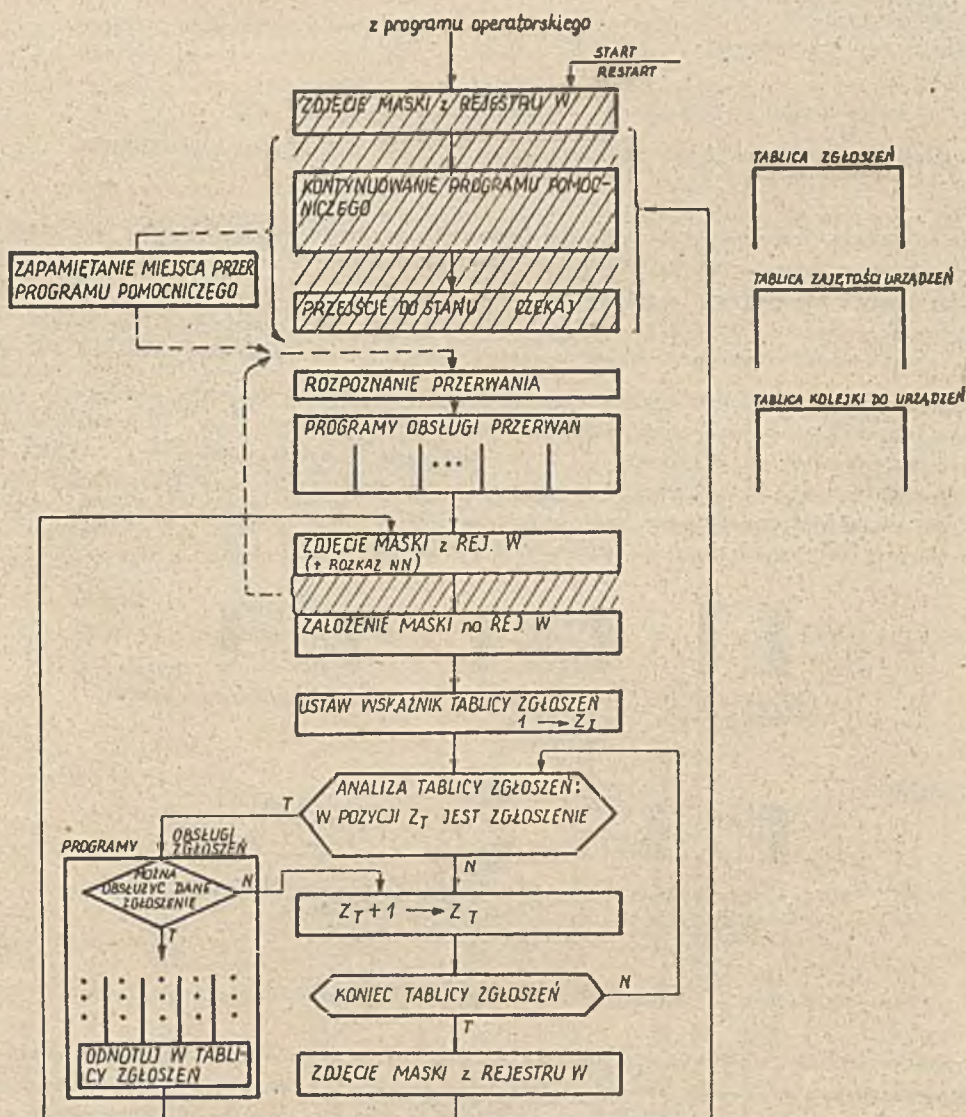
Rys. 18. Schemat struktury jednostek JSA1 i JSA2

Jednostki te zawierają z jednej strony, układy dopasowania do interfejsu minikomputera, z drugiej strony - zbiór rejestrów i bramek we - adresowalnych z minikomputera - umożliwiających przekazywanie poleceń (sygnałów sterujących) do urządzeń CP i odczytywanie stanu tych urządzeń. Ponadto układ generacji przerwań zapewnia wykrywanie zmian na określonych bitach informacji o stanie urządzeń i wysyłanie odpowiednich do tego sygnałów przerwań. Jednostki zawierają także odpowiednie układy transmisji sygnałów.

W celu przekazania bajtu danych do CP lub do innej jednostki SKSC-1, minikomputer instrukcją wo/wy (PISZ do...) wpisuje z szyn BY do adresatora jednostki numer rejestru wy. Następną instrukcją wo/wy wpisuje z szyn BY bajt danych do wybranego rejestru wy. Podobnie chcąc odczytać określony bajt stanu, minikomputer instrukcją wo/wy (PISZ do...) wybiera adresatorowi odpowiednią bramkę, a następną instrukcją wo/wy (CZYTAJ...) powoduje, że dane z tej bramki są podane na szynie BE. Jest to tzw. adresowanie dwustopniowe-pośrednie.

Dla realizacji opisanej wyżej organizacji przesyłania znaków POT jednostka JSM1 ma strukturę jak na rys. 19. Jednostka składa się z pięciu podkanałów oraz jednego układu sterowania podkanałami. W układzie sterowania podkanałami są dekodowane otrzymywane z minikomputera instrukcje oraz numer podkanału, do którego one się odnoszą. Umożliwia to m.in. programowy "start" i "stop" układu przesyłania znaków w podkanałach.

Gdy podkanał otrzyma sygnał "start", wyśle do kanału multipleksorowego sygnał R1 (zgłoszenie po znak). Po umieszczeniu tego znaku na szynach BY, podkanał otrzyma sygnał K1. Otrzymanie sygnału K1 spowoduje wpisanie znaku do rejestru R1. Pojawienie się lub obecność sygnału ZADANIE ZNAKU (z USN) spowoduje przepisanie znaku z R1 do R2 i wysłanie sygnału WEŻ ZNAK. Po przepisaniu znaku z R1 do R2 ponownie wysłany jest sygnał R1.



Rys. 20. Struktura oprogramowania podstawowego

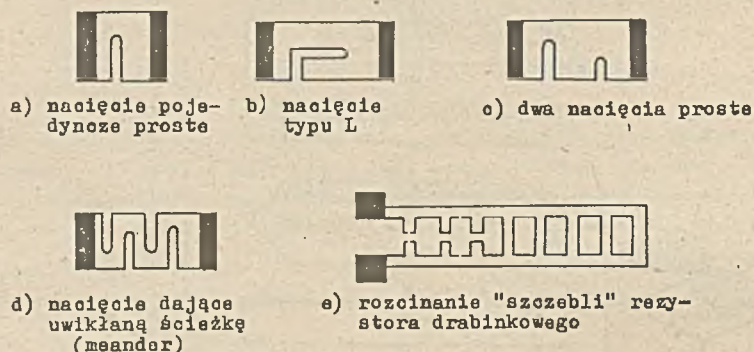
Zastosowanie zasady, że obsługa przerwania i obsługa jednego kroku zgłoszenia nie mogą być przerwane (maskowanie na ten czas rejestru W) znacznie upraszcza oprogramowanie, ponieważ nie tworzą się stosy przerw. Przyjęcie tej zasady oparte było na analizie wymagań czasowych, które CP narzuca SKSC-1 oraz na tym, że znaki POT przesyłane są przez kanał multipleksorowy, a dysk jest dołączony do kanału bezpośredniego dostępu.

Jeżeli program pomocniczy nie jest programem cyklicznym, to po jego zakończeniu następuje przejście komputera w stan "czekaj" (sygnalizowany lampką).

Laserowe urządzenia do automatycznej korekcji rezystorów warstwowych

W ostatnich latach przemysł elektroniczny wykorzystuje coraz więcej obwodów hybrydowych na podłożach płaskich z rezystorami warstwowymi. Dotychczas znane technologie (sitodruk do otrzymywania warstw grubych i naparowywanie dla warstw cienkich) nie pozwalają otrzymywać elementów o zadanej tolerancji oporności. Z tego powodu wykonuje się rezystory o mniejszej oporności, a następnie w odpowiedni sposób usuwa pewną część warstwy, korygując do wymaganej wartości rezystancji.

Najbardziej rozpowszechnionym sposobem korekcji jest nacinanie ścieżek, tak jak to pokazano na rys. 1a-d lub nacinanie "szczebli" rezystora w kształcie drabiny (rys. 1e). Znanych jest kilka metod usuwania warstwy rezystywnej [1]. W Przemysłowym Instytucie Elektroniki (CEMI-PIE) od paru lat wykorzystuje się metodę elektroerozyjną. Z przeglądu literatury technicznej ostatnich lat wynika, że najbardziej popularna jest metoda usuwania warstwy rezystywnej za pomocą promienia laserowego. Zalety jej są szczegółowo omówione w publikacjach [2], [3].



Rys. 1. Stosowane rodzaje nacięć przy korekcji rezystorów warstwowych

Należy zaznaczyć, że rozróżnia się dwa następujące rodzaje korekcji: pasywną i aktywną. Pierwsza z nich, najczęściej stosowana, polega na korekcji wg zadanej rezystancji. Ponieważ rezystory pełnią różne funkcje w obwodach elektronicznych, można je korygować w warunkach pracy danego obwodu wg zadanego parametru charakteryzującego realizowaną funkcję. Parametrem porównywanym lub badany może być tu częstotliwość, czas, napięcie lub prąd. Powyższy sposób nosi nazwę korekcji aktywnej lub funkcjonalnej.

• Rodzaje urządzeń

Urządzenia do korekcji rezystorów warstwowych można podzielić na cztery następujące grupy:

- sterowane ręcznie,
- częściowo zautomatyzowane,
- sterowane przez proste automaty,
- sterowane przez komputer.

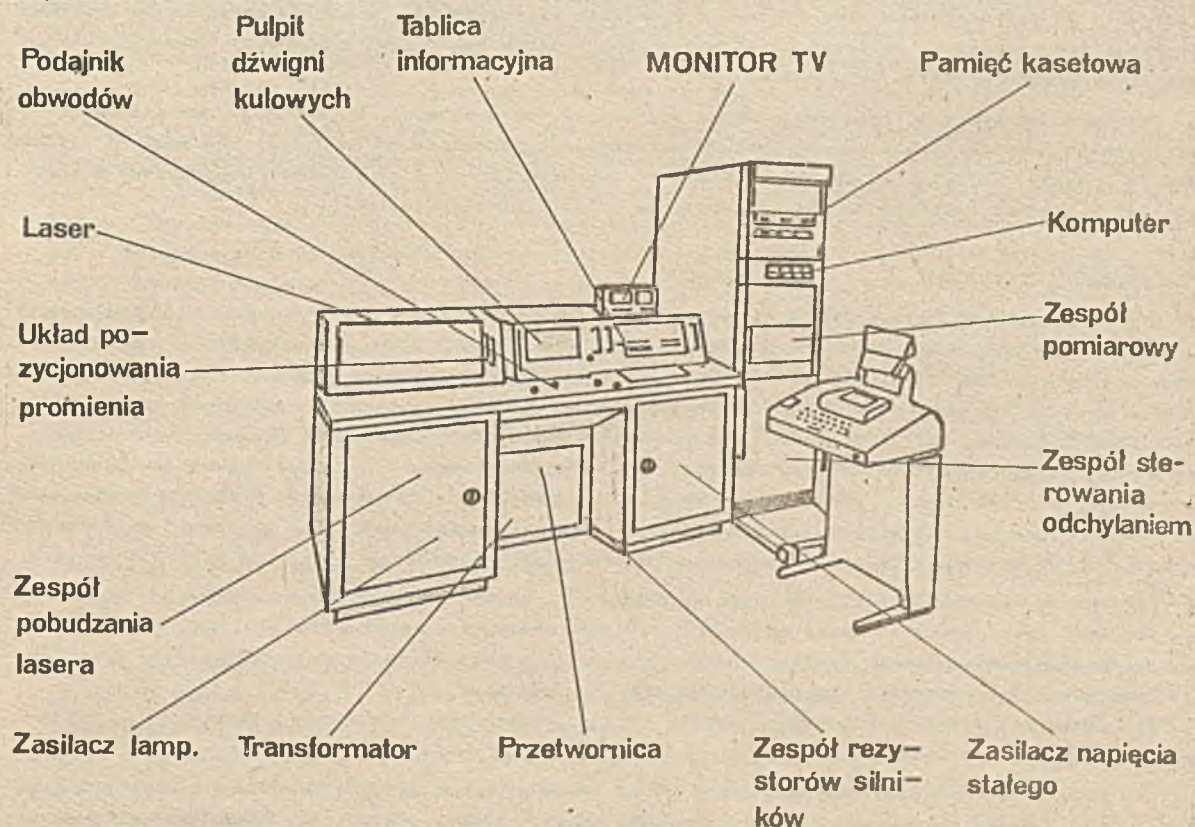
W urządzeniach pierwszej grupy podstawowe czynności korekcji tj. pozycjonowanie i nacinanie są wykonywane przez operatora. Urządzenia te wykorzystuje się przede wszystkim do prac laboratoryjnych. Są to najtańsze z urządzeń, ceny ich wahają się w granicach 25 + 30 tys. dolarów. Natomiast wadą ich jest udział człowieka we wszystkich czynnościach korekcji. Jest oczywiste, że wydajność tych urządzeń nie może być duża w porównaniu z automatami.

Do urządzeń częściowo zautomatyzowanych zalicza się takie, w których pewne operacje są wykonywane przez specjalne mechanizmy. Do tych ostatnich możemy zaliczyć automatyczny podajnik płytek

z rezystorami warstwowymi, programowany mechanizm przemieszczający płytkę w stosunku do nieruchomego promienia lub też przemieszczający promień po obrabianym rezystorze (w korekcie wykorzystującej laser), jak również układ automatycznie wyłączający promieniowanie w momencie uzyskiwania żądanej wartości rezystancji. Jest oczywiste, że za pomocą tego rodzaju urządzeń uzyskuje się większą wydajność korekty.

Dwie ostatnie grupy przyrządów charakteryzują się tym, że podstawowe operacje korekty, tj. pozycjonowanie, pomiar rezystancji i nacinanie odbywają się automatycznie i we wzajemnym powiązaniu. Urządzenia trzeciej grupy są wyposażone w komplet automatów programowanych ręcznie. W tym wypadku korekta jest wykonywana wg ustalonego szablonu ruchów, a rezystory korygowane porównywane są z rezystorami matrycy wzorcowej. Zmianę programu korekty dokonuje się albo przez odpowiednie ustawienie przełączników "programatora" albo przez wymianę specjalnej płyty krosowej. Urządzenia grupy trzeciej są do nabycia w cenie ok. 40 tys. dolarów.

Urządzenia czwartej grupy należą do najbardziej złożonych i jednocześnie najdroższych (od 75 do 115 tys. dolarów). Z drugiej strony odznaczają się wieloma zaletami w porównaniu z poprzednio omówionymi, dlatego też najszerszej będą omówione w niniejszym artykule. Ostatnio urządzeń tych, a raczej systemów buduje się coraz więcej. Są nawet urządzenia sterowane przez mikroprocesory wielkiej skali integracji. Na rys. 2 pokazano zestaw sterowanego komputerowo systemu do korekty rezystorów Model 80 firmy Micronetics.



Rys. 2. Zestaw sterowanego komputerowo automatycznego systemu korekty rezystorów

• Własności urządzeń sterowanych komputerowo

Do najważniejszych zalet sterowanych komputerowo urządzeń i systemów do korekty rezystorów należą:

- możliwość sterowania kilkoma urządzeniami do korekty przez jeden komputer,

- własność sterowania adaptacyjnego;
- możliwość prowadzenia różnego rodzaju statystyk oraz wyprowadzanie danych w różnych postaciach i formach.

Jak wiadomo jednostki centralne komputerów wykonują instrukcje ze stosunkowo dużą prędkością. Przy odpowiednim rozwiązaniu systemu korekcyjnego jeden komputer może obsługiwać kilka lub kilkanaście urządzeń korekcyjnych. Amerykańska firma TERADYNE oferuje system sterowany komputerowo, który może kierować pracą czterech stanowisk korekcyjnych.

Odpowiednio rozbudowany program sterujący korekcją może mieć własność wyciągania wniosków z bieżąco korygowanych rezystorów dla potrzeb efektywniejszego korygowania następnych rezystorów tego samego typu. Własność tę nazywamy sterowaniem adaptacyjnym.

Przy seryjnej produkcji płytek z rezystorami warstwowymi potrzebne są różnego rodzaju zestawienia danych, zarówno o wyrobach surowych (tj. przed korekcją), jak również o wyrobach skorygowanych, a także o wyrobach poddanych procesowi starzenia. Dane te są niezbędne do oceny jakości procesów technologicznych, użytych materiałów, a także dla oceny jakości korekcyjnej. Komputer może dostarczać oceny parametrów statystycznych jak również wykonywać różnego rodzaju raporty, a nawet wystawiać atesty dla płytek. Dane pomiarowe mogą być przedstawione w wartościach absolutnych, a także w liczbach względnych.

Jest możliwe rozwiązanie, w którym kodowanie danych dla nowego typu płytek odbywałoby się efektywnie za pośrednictwem tzw. systemu wspomagania komputerowego, w którym opracowanie programu korekcyjnego nie wymagałoby obliczeń konstrukcyjnych (program taki byłby generowany automatycznie na podstawie danych z reżimu korekcyjnego ręcznej).

Podsumowując powyższe informacje można powiedzieć, że urządzenia do korekcyjnych rezystorów ze sterowaniem komputerowym stanowią systemy uniwersalne. Natomiast chyba jedyną ich wadą są wysokie koszty nabycia. Czytelnikom zainteresowanym porównaniem urządzeń sterowanych komputerowo z pozostałymi rodzajami urządzeń do korekcyjnych polecam artykuł M. Nesselrotha [4].

• Wydażność urządzeń

Z punktu widzenia potencjalnego nabywcy dość ważnym parametrem jest wydajność urządzenia bądź systemu. Mimo licznych wzmianek producentów niezbyt dokładnie precyzują wydajność lub produktywność swoich wyrobów. W artykułach i notatkach w czasopiśmie technicznych czy w prospektach firmowych spotyka się parametry, które powinny pozwolić na ocenę wydajności tych urządzeń. Jednakże wobec braku ścisłej definicji producentów podają parametry te wg własnej i najczęściej sobie tylko znanej interpretacji. Stąd mimo pozorów, trudno jest porównywać wydajność korekcyjnych urządzeń wytwarzanych przez różnych producentów. Wydajność korekcyjnej zależy od wielu, niekiedy wzajemnie ze sobą powiązanych czynników. Pominięcie ich w prospektach firmowych może prowadzić do fałszywej oceny różnych systemów. Do najważniejszych czynników wpływających na wydajność korekcyjnej należą:

- prędkość wprowadzania płytek na pole obróbki,
- prędkość pozycjonowania rezystorów,
- prędkość przemieszczania płytek, promienia laserowego lub narzędzia obrabiającego, między dwoma kolejnymi operacjami korekcyjnymi,
- czas pomiaru rezystancji,
- wymagana tolerancja rezystancji,
- rodzaj i liczba nacisków,
- liczba rezystorów na jednej płytce oraz ich rozmiary,
- gęstość liniowa energii lasera,
- rozwiązanie systemu komputerowego.

Jak podano wcześniej, w urządzeniach z grupy trzeciej i czwartej, a także w urządzeniach grupy drugiej, płytki z rezystorami są wprowadzane na pole obróbki za pomocą automatycznych podajników. Najczęściej płytki obwodów hybrydowych mocuje się w odpowiedni sposób na specjalnych płytach o kształcie pierścienia lub prostokąta. Z kolei płyty przemieszcza się pod wpływem odpowiedniego sterowania. Istnieją podajniki, które wprowadzają płytki (kolejno) na pole obróbki w czasie krótszym od 1 sekundy. Natomiast czas wymiany płytek oraz przełączania sond kontaktowych ok. 1 minuty zalicza się do bardzo krótkich (System 25 firmy ESI). Pozycjonowanie wymaga specjalnych układów sterujących i rozpoznających charakterystyczne punkty płytki lub rezystora. Firma TERADYNE wykorzystuje metodę, w której czas pozycjonowania i czas przemieszczania między operacjami dwu

kolejnych rezystorów wynosi 10 ms. Przy odpowiednim rozwiązaniu mocowania płytek z rezystorami do płyty podajnika, można uniknąć operacji pozycjonowania.

Jest oczywiste, że mechanizm przemieszczający promień laserowy lub narzędzie obrabiające w stosunku do płytki (albo odwrotnie, mechanizm przemieszczający płytkę w stosunku do promienia lub narzędzia) powinien być skonstruowany pod kątem uzyskania najniższego upływu czasu między kolejnymi korektami. W metodzie z przemieszczaniem promienia laserowego za pomocą ruchomych zwierciadeł - uzyskuje się stosunkowo duże prędkości, umożliwia to mała inercja elementów przemieszczanych, tj. lustro i wałek mechanizmu. I tak przy zastosowaniu urządzenia wybierającego G300PD [5] Optical Scanner z lustrem o wymiarach 38 x 50 mm i bezwładności 18,7 gcm² można uzyskać odchylenie promienia o 26° w czasie 8 ms, co przy odległości osi obrotu lustro od powierzchni obrabianej równej 2,4 cm daje przemieszczenie promienia po tej powierzchni o około 1,1 cm. Dla porównania można przyjąć, że maksymalna prędkość dla metody z przemieszczaniem luster może wynosić 135 cm/s, podczas gdy maksymalne prędkości przemieszczania stołów koordynatowych (stosowanych w systemach korekcyjnych rezystorów) zawierają się w granicach od 1 do 4 cm/s.

Nacinanie rezystorów jest procesem ciągłym, Upływ czasu między momentem rozpoczęcia pomiaru rezystancji a momentem wyłączenia nacinania powinien być krótszy od dopuszczalnego przyrostu oporności dla danego upływu czasu. Wynika z tego, że dla potrzeb korekcyjnych należy budować mierniki rezystancji o bardzo krótkim czasie pomiaru. Jednakże mierniki o krótkim czasie pomiaru są mniej dokładne. Zatem im większą chcemy uzyskać dokładność tym wolniej należy przeprowadzać nacinanie. Z tych i jeszcze innych względów [6] prędkość nacinania warstwy odbywa się z o wiele mniejszą prędkością od maksymalnej. Spotykane wartości prędkości wynoszą: dla urządzenia AZOTI prod. ZSRR 0,2 cm/s (przy dokładności korekcyjnej do 0,5%) oraz dla urządzenia firmy DU PONT [6] 0,5 + 10 cm/s.

W większości wypadków czas korekcyjny będzie dłuższy dla nacięć o kształcie litery L lub dla nacięć wielokrotnych, w porównaniu z nacięciami o kształcie litery I (prostymi). Jednakże nacięcia o złożonym kształcie lub wielokrotnie stosuje się dla uzyskania większej stabilności czasowej lub też dla uzyskania większej wytrzymałości napięciowej. Z punktu widzenia zależności prędkości korekcyjnej od gęstości upakowania rezystorów na płytce, można uważać, że średni czas korekcyjny będzie krótszy dla płytek zawierających większą liczbę rezystorów w porównaniu z płytkami o tych samych rozmiarach ale o mniejszym upakowaniu. Przy większym upakowaniu na ogół są mniejsze odległości między rezystorami, stąd mniejsze straty na przemieszczanie bez korekcyjnej. Zagadnienie wpływu gęstości liniowej energii lasera jest omówione w publikacji [6]. Wydajność korekcyjnej zależy od urządzeń składowych systemu komputerowego, jak również od przyjętego oprogramowania. Np. przy zbyt wolnej transmisji między urządzeniami może wystąpić znaczne wydłużenie czasu korekcyjnego lub zmniejszenie dokładności. Czas wyprowadzania danych oraz czas wprowadzania nowego programu korekcyjnego może być źródłem dodatkowych opóźnień.

Wyżej wymienione czynniki istotnie wpływają na prędkość korekcyjną. Wynika z tego, że poszczególne typy urządzeń można by było porównywać tylko wtedy gdyby parametr ten był podawany dla ustalonej dokładności korekcyjnej, kształtu i liczby nacięć oraz dla standardowej płytki o ustalonej liczbie rezystorów, ustalonym ich rozmieszczeniu i wymiarach.

W rzeczywistości najczęściej spotykanymi parametrami oceny wydajności są: liczba nacięć korekcyjnych w jednostce czasu, liczba płytek skorygowanych w jednostce czasu lub też czas korekcyjny i rezystora. Należy dodać, że w większości wypadków producenci ograniczają się do podania tylko jednego z ww parametrów. Dość rozpowszechnione jest podawanie liczby nacięć w ciągu godziny. Przy takim sformułowaniu niewiele można powiedzieć o wydajności, a to z tego powodu, że producenci nie podają ani dokładności rezystancji ani liczby rezystorów na płytce. Zakres spotykanych wartości tego parametru dla urządzeń korygujących promieniem laserowym zawiera się w granicach od 900 (Lasertrim firmy APOLLO LASERS) do 10000 (KRTM firmy KORAD) nacięć na godzinę. Natomiast urządzenia korygujące metodą piaskowania z prędkością 1500 (Model RT-10SA firmy MPM Corp.) nacięć na godzinę, zalicza się do bardzo szybkich. Wyjątkowo dokładne dane podają autorzy publikacji [1] a mianowicie, że opracowany w Instytucie Manfreda (NRD) system do korekcyjnej rezystorów metodą wiązki elektronowej może korygować z prędkością 20 do 40 tys. nacięć na godzinę z dokładnością do 1% dla płytek o wymiarach 60 x 100 mm, zawierających po 10 rezystorów.

Firma TERADYNE posługuje się drugim z ww parametrów, tj. liczbą płytek skorygowanych na godzinę, który dla urządzenia W233 wynosi 3000. Również i ta ocena nie jest dokładna wobec braku informacji o dokładności rezystancji, liczbie rezystorów, na płytce, kształcie nacięć itd. W prospekcie zautomatyzowanej wersji urządzenia KRT firmy KORAD podany jest średni czas korekcyjnego jednego rezystora do tolerancji 0,1%, który wynosi 0,5 sekundy. Natomiast brak jest tu wielu innych danych.

Przy braku jednoznacznych informacji najbardziej wiarygodne dane o wydajności danego urządzenia można otrzymać od producenta, a mianowicie przez zażądanie określenia przez niego czasu korekcyjnego ustalonej partii reprezentacyjnej płytek (określonej przez pytającego).

Przegląd produkowanych urządzeń

W tabelach 1 i 2 przedstawiono parametry techniczne i inne dane urządzeń czwartej i trzeciej grupy, tj. odpowiednio urządzeń ze sterowaniem komputerowym oraz urządzeń sterowanych przez automaty proste.

Na uwagę zasługują urządzenia amerykańskiej firmy TERADYNE. Są to raczej zestawy urządzeń składające się z kilku oddzielnych jednostek konstrukcyjnych (modułów), które dzięki niezmiennym od kilku lat interfejsom dają się zestawiać w różne konfiguracje o mniejszym lub większym stopniu zautomatyzowania. Przejście do systemu bardziej zautomatyzowanego polega na uzupełnieniu istniejącego zestawu dodatkowymi modułami. Zestaw ze sterowaniem ręcznym typu W311M zawiera moduły podstawowe. Można go rozbudować do systemu sterowanego przez automat (prosty), programowany ręcznie W233 (zob. tabela 2) bądź do systemu jedno- lub nawet czterostanowiskowego W311 (tabela 1) sterowanego przez komputer.

Zorientowany produkcyjnie system W311 o nazwie Dynamic Laser Adjustment System jest dostępny w czterech wersjach, przy czym trzy z nich są przeznaczone do korekcyjnej pasywnej, a wersja czwarta do korekcyjnej aktywnej, według zadanej częstotliwości. Wersje do korekcyjnej pasywnej różnią się bądź sposobem podawania płytek obwodów hybrydowych na pole obróbki wersja z podajnikiem sterowanym ręcznie albo wersja z podajnikiem sterowanym automatycznie bądź sposobem korekcyjnym (podajnik wykonujący ruchy z powtórzeniami).

Procesor części komputerowej składa się ze specjalizowanego sterownika M365 oraz pamięci operacyjnej o pojemności 12 kilobajtów 18-bitowych. W miarę potrzeby pamięć może być uzupełniana modułami o pojemności 4 kilobajty. Do wersji podstawowej systemu komputerowego należy jeszcze terminal (zawierający alfaskop i klawiaturę) oraz pamięć kasetową. Opcyjnie konfiguracja sprzętowa może być uzupełniona o pamięć taśmową, drukarkę wierszową oraz monitor TTY. Oprogramowanie podstawowe steruje korekcją płytek zawierających jeden lub więcej rezystorów warstwowych. Możliwe jest tu wykonywanie nacięć o kształcie filtry L lub nacięć wielokrotnych dla każdego rezystora. Pod wpływem sterowania komputerowego oraz pomocniczego lasera, system szybko wykrywa krawędź rezystora przy przemieszczaniu układu pozycjonującego między korekcjami dwóch kolejnych rezystorów, co wydajnie zwiększa prędkość korekcyjną. Oprogramowanie podstawowe wyprowadza tzw. arkusze zestawień zawierające wyniki pomiarów wstępnych oraz pomiarów po korekcji. Programy systemu W311 mają własności adaptacyjne, polegające na tym, że dane z korekcyjnej wcześniejszej obróbki płytek są wykorzystywane jako bazy dla następnych programów nacinania. Dla korekcyjnej z powtórzeniami oraz dla korekcyjnej aktywnej stosuje się specjalne programy użytkowe. Producent oferuje, jako opcje, programy takie jak: sterowanie mocą lasera, ocena wpływu starzenia się i temperatury, sterowanie ciepłem płytek podłożonych i inne.

W systemie wielostanowiskowym każde stanowisko korekcyjne zawiera własny laser, układ pozycjonujący promień, układ sterowania laserem, zestaw TV (do oglądania elementów obrabianych) oraz odpowiedni podajnik płytek obwodów hybrydowych. Natomiast zestaw komputera i mostek rezystancji są wspólne dla całego systemu.

W tabeli 2 oprócz wyżej opisanego przedstawiono parametry trzech innych systemów sterowanych przez komputer. Są nimi: system Model 685 firmy LASER OPTRONIC, system 25 firmy ELECTRO SCIENTIFIC INDUSTRIES oraz system Model 1080 firmy QUANTRAD i system 25 firmy ELECTRO SCIENTIFIC INDUS-

TRIES. Należy nadmienić, że w publikacji [2] zaprezentowano wyniki doświadczeń z zakresu korekacji tym systemem.

Wyżej wymieniony system firmy QUANTRAD (USA) jest sterowany przez mikroprocesor*. System jest wyposażony w pamięć operacyjną o pojemności 24 kilobajtów (typu RAM). W miarę potrzeby może ona być powiększona do 56 kilobajtów. Pozostałymi zespołami części komputerowej są: podwójny dysk elastyczny oraz monitor ekranowy z klawiaturą. Monitor ten może być wykorzystywany jako monitor TV do oglądania płytek podczas korekacji. System komputerowy jest uwarunkowany czasowo (praca w czasie rzeczywistym). Wprowadzanie danych początkowych i innych informacji w czasie procesu korekacji odbywa się na zasadzie dialogu komputera z operatorem. Program może żądać od operatora zmiany wartości rezystancji lub położenia promienia laserowego. System dysponuje środkami dla programowania interakcyjnego i adaptacyjnego, które poważnie skracają czas pisania programów. Oprogramowanie wewnętrzne tłumaczy informacje pisaną w języku interakcyjnym, jak również jest przystosowane do obsługi standardowych interfejsów. Istnieje możliwość sterowania korekcją aktywną. Ponadto w systemie pracuje program testujący zespół lasera, który w razie niesprawności informuje odpowiednio operatora, wyświetlając parametry takie jak: moc i temperaturę wyłącznika Q, temperaturę wody, stan blokad itp. System ma zdolność korygowania rezystorów grubo- i cienkowarstwowych do tolerancji w zakresie od 0,1 do 0,001%.

Urządzenia sterowane przez automaty proste może podzielić na dwa rodzaje: programowane za pomocą zespołu przełączników oraz programowane przez wymianę płytek kontaktowych. Spośród wymienionych w tabl. 2, urządzeniami pierwszego rodzaju są W233 - Unitrim firmy TERADYNE oraz Lasertrim firmy APOLLO LASERS. Pozostałe dwa typy tj. KRT firmy KORAD oraz AZOT i produkcji ZSRR programuje się przez wymianę płytek.

Urządzenie W233 firmy TERADYNE jest wyposażone w urządzenie programujące, na którym można ustawić następujące parametry korekacji: długość i kierunek nacinania, wartość rezystancji (lub napięcia), prędkość pracy wyłącznika Q lasera a także prędkość napinania. Urządzenie może korygować z prędkością do 1000 płytek na godzinę przy wykorzystaniu podajnika automatycznego, może również wykonywać nacięcia pojedyncze, podwójne o kształcie litery L a także w kształcie serpentyny.

Podobnie jak TERADYNE firma KORAD oferuje urządzenia sterowane ręcznie, częściowo zautomatyzowane oraz urządzenia sterowane przez automaty proste, natomiast brak jest informacji o systemach sterowanych przez komputer. Oprócz urządzenia KRT wyszczególnionego w tabl. 2, istnieje bardziej wydajne urządzenie firmy KORAD o nazwie KRTM, które różni się od poprzedniego obecnością podajnika karuzelowego i nowym typem programowanego urządzenia pozycjonującego. W poprzednim punkcie podano prędkość korekacji tego urządzenia.

Pozostałe urządzenia z tabl. 2 nie wymagają specjalnego omówienia ze względu na wystarczająco szczegółowe informacje podane w tej tabeli.

Prace prowadzone w Polsce

W Polsce prowadzone są prace nad zbudowaniem automatycznych urządzeń do korekacji rezystorów warstwowych przy pomocy promienia laserowego, a mianowicie w Instytucie Budowy Sprzętu Precyzyjnego i Elektronicznego Politechniki Warszawskiej oraz w Zakładzie Doświadczalnym Radiokomunikacji Gdynskich Zakładów Radiowych RADMOR. Wyniki prac osiągniętych przez konstruktorów w ZD RADMOR przedstawiono w pracy [8]. W 1976 r. w Instytucie Maszyn Matematycznych w Warszawie opracowano projekt systemu jedno- i wielostanowiskowego sterowanego przez komputer. Ponadto należy nadmienić, że w Przemysłowym Instytucie Elektroniki Naukowo-Produkcyjnego Centrum Półprzewodników w Warszawie opracowano i wykonano kilka urządzeń do korekacji rezystorów cienkowarstwowych, wykorzystujących metodę elektroerozyjną. Należy zaznaczyć, że urządzenia te są sterowane ręcznie.

* Notatka w miesięczniku Electronic Packaging and Production z marca 1977 r. str. 53.

Tabl. 1

PARAMETRY URZĄDZEŃ DO KOREKCJI REZYSTORÓW GRUBO- I CIENKOWARSTWOWYCH STEROWANYCH KOMPUTEROWO

| Lp. | Typ urządzenia Produce | Typ komputera lub sterownika | Pojemność PAO | Pamięć i urządzenia zewnętrzne | Charakterystyka oprogramowania | Metoda korekcji i inne dane techniczne | Maksymalne rozmiary płytek obrobionych mm | Zakres wartości rezystancji | Osiągana dokładność rezystancji | Rodzaj podajnika płytek | Cena w dolarach |
|-----|---|---------------------------------|---|---|---|--|--|-----------------------------------|---------------------------------------|---|--------------------------------|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 1. | Model 311 Dynamic Laser Adjust System TERADYN (USA) | M365 Controler | 12 kół (moduły opcjonalne 4 kół wa 18-bitowe) | Alfaskop z klawiaturą Pamięć kasetowa. Opcje: pamięć taśmowa, drukarka, TTY | Programy optyczne: sterowanie modułem lasera: kontrola z powtórzenia korekcja wg częstotliwości Ocena starzenia i wpływu temperatury | Laser YAG (z przełącznikiem Q). Odchylenie promienia za pomocą specjalnego urządzenia pozycjonującego. System komputerowy może sterować ztemperatura stanowiącymi korekcji | 10,2x10,2 24,4x25,4 | od 10 Ω od 20 MΩ | 0,1 | Pierścieniowy rezerwuuar Pierścieniowy autotomator Liniiowy z powtarzaniem Stół specjalny do korekcji aktywnej | 97.500* 112.500* 88.500* |
| 2. | Model 685 LASER OPTRONIC | Komputer H-P serii 2100 | | Monitor TTY | Specjalny język ułatwiający programowanie korekcji | Laser YAG Ruchomy system optyczny do przeszacowania promienia. Koryguje płytki zawierające do 40 rezystorów. Może sterować sześcioma stanowiącymi. | | | | Stół obrotowy | |
| 3. | System 25 ELECTRO-SCIENTIFIC INDUSTRIES (USA) | | | Alfaskop z klawiaturą | | Laser Promień lasera jest ruchomy. Dokładność pozycjonowania 2,5 m | | do 1 GΩ | 0,005 | | |
| 4. | Model 1080 QUANTRAD CORPOR (USA) | Mikrokomputer | (RAM) 24 kbajt. a przy opcji do 56 kbajtów | Monitor ekranowy z klawiaturą. Podwójny dysk elastyczny | Język do programowania interakcyjnego i adaptacyjnego. Sterowanie w czasie rzeczywistym. Dialogowa komunikacja programu z operatorem. | Laser YAG (z przełącznikiem Q). Odchylenie promienia za pomocą galvanometrycznego urządzenia pozycjonującego. Pomiar rezystancji za pomocą mostka. Monitor systemu może być wykorzystywany jako monitor TV do oglądania rezystorów korygowanych. | 50,8x50,8 | | od 0,1 do 0,001 | Stół obrotowy 6-podajnik z podajnikiem autotomator | 85.000 (1977 r.) |

*) Ceny z 1973 r. dla systemu jednostanowiskowego.

Tabl. 2

PARAMETRY URZĄDZEŃ DO KOREKCJI REZYSTORÓW GRUBO- I CIENKOWARSTWOWYCH STEROWANYCH PRZECZ AUTOMATY PROSTE

| Lp. | Typ urządzenia Producent | Metoda korekcji i inne dane techniczne | Maksymalne wymiary płytek obrabianych mm | Zakres wartości rezystancji | Osiągana dokładność rezystancji % | Makymalna prędkość mieszczania stołu | Rodzaj podajnika płytek | Parametry pozwalające oszacować wydajność korekcji | Parametry programowane | Cena w dolarach |
|-----|---|---|---|-----------------------------|--------------------------------------|--|---|--|--|-----------------|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 1. | W233-Unitrimmer THERADYNE (USA) | Laser YAG (z przełącznikiem Q). Odechnianie promienia za pomocą galwanometrycznego urządzenia pozycjonującego. Wyposażony w programowane urządzenie sterujące: programowanie za pomocą przełączników. | 10,2x10,2 25,4x25,4 50,8x50,8 | od 10Ω do 20MΩ | 0,1 | | Pierścieniowy ręczny lub automatyczny. Liniowy do pracy z powtórzeniami | do 3000 płytek na godz. 1) Przemieszczanie promienia 2) Wartość rezystancji 3) Częstotliwość powtórzeń impulsów lasera 4) Prędkość nacinania | | |
| 2. | KRT System KORAD (USA) | Laser YAG (z przełącznikiem Q). Automatyczne odechnianie promienia. Wyposażony w programowany mostek oraz programowane urządzenie pozycjonujące z matrycą oporową | 50,8x50,8 | od 0,01Ω do 111MΩ | | | | Średni czas korekcji rezystora do tolerancji 0,1%; 0,5s | 1) Przemieszczanie promienia 2) Wartość rezystancji | 40.000 |
| 3. | AZOT 1 prod. ZSRP | Laser λ_2 - impuls. Przemieszczanie płytek zamontowanych na stole koordynatowym. Wyposażony w programowane urządzenie sterujące i mostek rezystancji | 40 x 60 | od 60Ω do 6N | 0,5 | do 0,2cm/s przy naciśnięciu; do 1 cm/s przy ruchu jałowym | | | 1) Przemieszczanie stołu 2) Wartość rezystancji | |
| 4. | Lasertrim [®] APOLLO LASERS (USA) | Laser CO ₂ - impuls. Przemieszczanie poje-dynczych płytek zamontowanych na stole koordynatowym. Wyposażony w urządzenie sterujące - Model 400 | | | | | Podajnik automata-tyczny typu próżniowego | do 900 płytek na godz. Średni czas korekcji rezystora 1 s. | 1) Przemieszczanie stołu 2) Wartość rezystancji | |

*) Koryguje rezystory grubowarstwowe i wielkość rezystorów cienkowarstwowych

Literatura

- [1] SCHILLER S., MEISING V., PANZER S.: Electron.-Beam Trimming of Thin and Thick Film Resistor Networks. Solid State Technology 1975 vol. VII
- [2] CADENHEAD R.L.: Production Trade-Offs in Laser Trimming Operations Solid State Technology 1976 vol. I
- [3] DRGZD Z., LEMANOWICZ J.: Obróbka laserowa w mikroelektronice. Materiały sekcji 2, III międzynarodowej konferencji naukowo-technicznej "Mikronika 77", Warszawa, 29-30.11.1977 r.
- [4] NESSELROTH M.: Laser Trimming: Manual or Computer - Controlled. Electronic Packaging and Production 1974 nr 9
- [5] Katalog firmy OPTELEC, Monachium - RFN, 1975
- [6] Laser Trimming Techniques for Thick Film Resistors. Prospekt firmy E.I. DU PONT DE NEMOURS, 1973
- [7] Prospekt firmy TERADYNE, rozdział pt. "Laser Trimming Equipment"
- [8] PETRYKOWSKI Z.: Tester rezystancji mikroukładów cienkowarstwowych VA-11. Materiały sekcji 3, III międzynarodowej konferencji naukowo-technicznej "Mikronika 77", Warszawa, 29 - 30.11.1977

mgr inż. Jan WRONA

Biuro Generalnych Dostaw CNPTKIP
Pracownia Projektowa

Organizacja bezpośredniego przesyłania informacji w obiektowych systemach sterowania

Wstęp

Standardowe komputery pozwalają na bezpośrednie czytanie informacji z odległości nie przekraczającej kilkunastu metrów. Dla umożliwienia czytania informacji z dalszych odległości stosuje się specjalne rozwiązania, których zasada opiera się na pracy asynchronicznej w stosunku do cyklu wewnętrznego komputera. Wszystkie stosowane dotychczas rozwiązania można sprowadzić do jednego schematu polegającego na tym, że informacja z obiektu przesyłana jest do rejestru zewnętrznego, przy czym każdemu obiektowi przyporządkowany jest odpowiedni rejestr przy lub w komputerze, a następnie jest czytana do komputera.

Zasadniczą wadą takich rozwiązań jest konieczność zastosowania dużej liczby dodatkowych elementów logicznych związanych z przesyłaniem i rejestracją informacji.

Poniżej będzie przedstawiony sposób rozwiązania czytania na znaczne odległości w czasie jednego cyklu wewnętrznego instrukcji "czytaj" - dla komputera ze specjalną organizacją wewnętrzną lub w czasie dwóch cykli "czytaj" dla standardowego komputera wyposażonego w specjalną przystawkę.

Istota rozwiązania

Istota proponowanego rozwiązania polega na wprowadzeniu, oprócz istniejącej szyny informacyjnej łączącej komputer z obiektem zewnętrznym - szyny służącej do przesyłania informacji do i z obiektu oraz szyny adresowej dodatkowych dwóch przewodów, z których jeden służy do przesyłania z komputera do obiektu zewnętrznego sygnału "CZYTAJ I", a drugi do przesyłania sygnału "CZYTAJ II" w kierunku odwrotnym, przy czym sygnał "CZYTAJ II" stanowi odpowiedź obiektu na sygnał "CZYTAJ I". Sygnał "CZYTAJ II" powoduje wpisanie informacji przesyłanej z obiektu do specjalnego rejestru, który jest wspólny dla wszystkich obiektów. Czytanie informacji z wszystkich obiektów zewnętrznych objętych układem czytania na odległość odbywa się za pośrednictwem jednego numeru wejścia/wyjścia, któremu przyporządkowany jest rejestr zewnętrzny, traktowany przez komputer jako urządzenie peryferyjne.

Odczytywanie informacji w proponowanym rozwiązaniu wymaga przesyłania do obiektu zewnętrznego jego adresu oraz sygnału "CZYTAJ I". Wraz z tymi sygnałami wysyłany jest szyną informacyjną kod instrukcji "CZYTAJ" z wydzielonej komórki pamięci komputera bądź specjalnego układu.

Obiekt po odebraniu tych sygnałów wysyła sygnał "CZYTAJ II" oraz szyną informacyjną żądaną informację. Sygnał "CZYTAJ II" wpisuje informacje w rejestr, który jest traktowany - z wcześniejszymi ustaleniami - jako urządzenie zewnętrzne. Następnie informacja zostaje prze czytana do komputera.

Warunkiem prawidłowej pracy układu jest zachowanie takiej maksymalnej odległości między komputerem a obiektem zewnętrznym, aby czas przejścia sygnałów od komputera do obiektu i z powrotem był mniejszy od czasu między wysłaniem sygnału "CZYTAJ I" a przeczytaniem informacji z rejestru do komputera.

Zastosowanie rozwiązania będzie pokazane poniżej na dwóch przykładach.

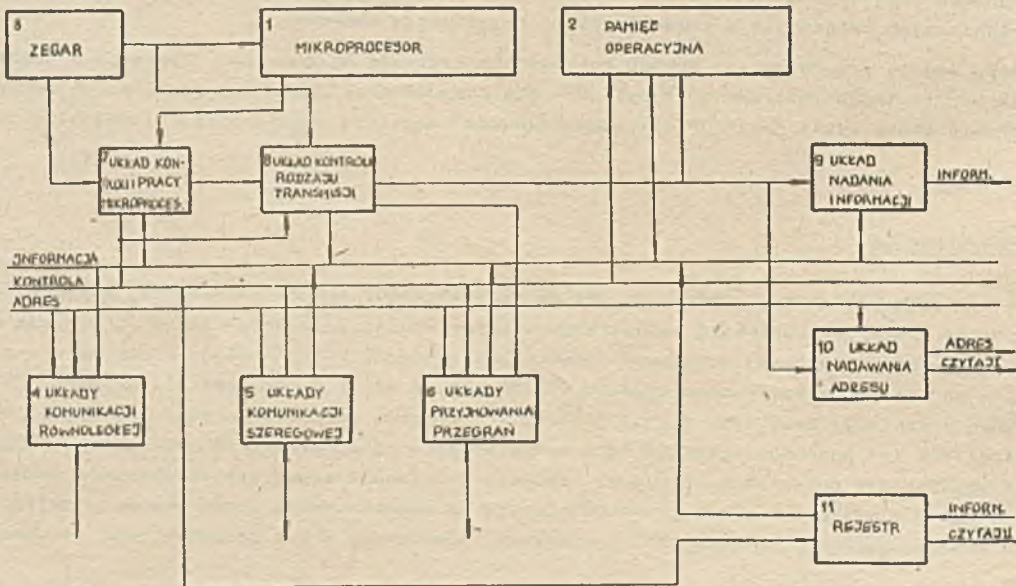
• PRZYKŁAD I - Organizacja wewnętrzna minikomputera zbudowanego na mikroprocesorze

Przedstawiony poniżej minikomputer oprócz standardowej komunikacji za pośrednictwem układów

do transmisji równoległej i szeregowej wyposażony jest w układ do transmisji równoległej na znaczne odległości.

Przyjęto, że każdemu urządzeniu - w tym rodzaju komunikacji - odpowiada ściśle określony numer komórki w pamięci operacyjnej. Jeżeli więc w niej jest zapisany rozkaz wejścia/wyjścia, to znaczy, że rozkaz odnosi się do danego urządzenia zewnętrznego.

Na rys. 1 pokazana jest budowa blokowa opisywanego minikomputera. Standardowy minikomputer zbudowany na mikroprocesorze składa się z: mikroprocesora (1), pamięci operacyjnej (2), zegara (3), układu do standardowej równoległej komunikacji (4), układu do standardowej szeregowej komunikacji z urządzeniami zewnętrznymi (5), układu przyjmowania przerwań zewnętrznych (6) oraz układu kontroli pracy mikroprocesora (7), spełniającego rolę synchronizacji i wybierania odpowiedniego rodzaju współpracy z mikroprocesorem. Dodatkowo wprowadzone bloki to przede wszystkim układ kontroli rodzaju transmisji (8), którego zadaniem jest sprawdzenie rodzaju transmisji. Jeżeli okaże się, że rozkaz wejścia/wyjścia jest zapisany w komórkach pamięci operacyjnej, których numery są przewidziane jako numery urządzeń zewnętrznych, numer komórki operacyjnej jest wysyłany z układu nadawania (10) jako numer urządzenia zewnętrznego wraz z sygnałem "CZYTAJ I". Jednocześnie za pomocą specjalnego układu informacyjnego (9) z wydzielonej komórki pamięci operacyjnej zostaje wysłany na szynę informacyjną kod instrukcji czytaj.



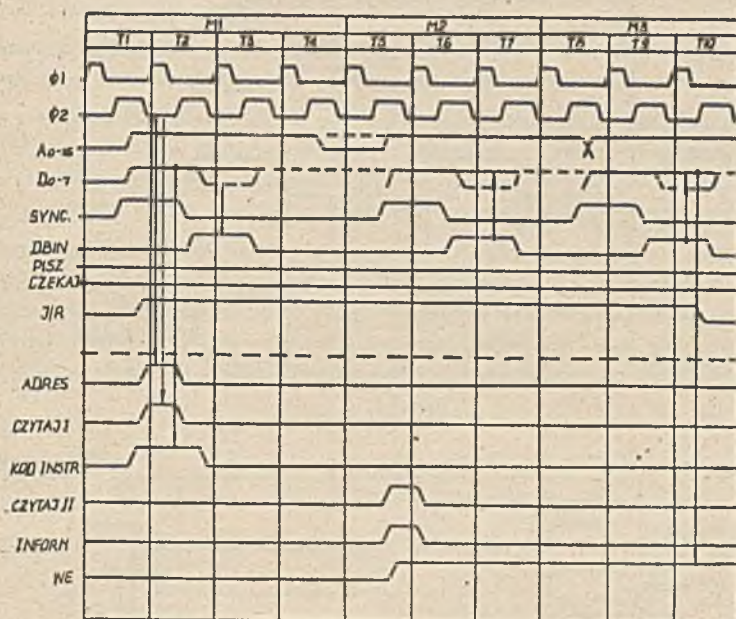
Rys. 1.

Informacja wysłana przez obiekt zewnętrzny przychodzi odpowiednią szyną informacyjną i jest wpisywana do rejestru (11) sygnałem "CZYTAJ II" generowanym przez obiekt zewnętrzny, skąd jest czytana do mikroprocesora.

Ponadto układ kontroli transmisji zgłasza przerwania programowe do mikroprocesora za pośrednictwem układu przyjmowania przerwań zewnętrznych (6) wówczas, gdy do chwili odczytania informacji z rejestru (11) do mikroprocesora nie nadszedł sygnał "CZYTAJ II".

Jeżeli rozkaz wejścia/wyjścia jest zapisany poza wydzieloną grupą komórek operacyjnych przewidzianych jako numery urządzeń zewnętrznych, układ kontroli transmisji (8) powoduje włączenie standardowych układów transmisji (4) lub (5).

Na rys. 2 przedstawione są przebiegi związane z czytaniem informacji w minikomputerze opartym na mikroprocesorze. Nad linią kreskową są przedstawione przebiegi operacji przy wykonywaniu przez mikroprocesor standardowej instrukcji "CZYTAJ" dla obiektów i urządzeń zewnętrznych, natomiast pod linią - przebiegi wykonywania instrukcji "CZYTAJ" dla obiektów znacznie oddalonych.



Rys. 2.

Cykl mikroprocesora trwa dziesięć cykli zegarowych T . W czasie pierwszego cyklu zegarowego T_1 zostaje ustawiony licznik rozkazów A_{0-15} oraz pobrana informacja D_{0-8} z komórki pamięci operacyjnej określonej licznikiem rozkazów A_{0-15} ; jeżeli w komórce pamięci jest zapisany rozkaz "CZYTAJ", zapala się przerzutnik w dekodерze rozkazów - sygnał J/R . Jeżeli układ kontroli transmisji (8) stwierdzi, że numer komórki pamięci (A_{0-15}) odpowiada numerowi urządzenia zewnętrznego, sygnał zegarowy Φ_2 jest wysłany jako sygnał "CZYTAJ I" i jednocześnie adres komórki A_{0-15} w iloczynie z Φ_2 jako adres urządzenia zewnętrznego. Jednocześnie zostaje wysłana zawartość wydzielonej komórki pamięci w iloczynie z sygnałem SYN jako kod instrukcji "CZYTAJ". Po przyjęciu z obiektu zewnętrznego informacja będzie wpisana do rejestru sygnałem "CZYTAJ II", skąd zostanie przeczytana w określonym cyklu zegarowym do komputera.

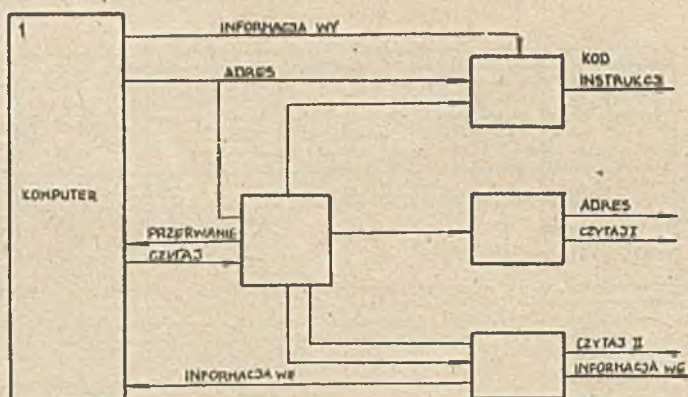
● PRZYKŁAD II - Budowa przystawki umożliwiającej komunikację na dalsze odległości

W kanale znakowym specjalnej przystawki do czytania na odległość organizacja jest podobna do organizacji przedstawionej w przykładzie I. W grupie adresów należy wydzielić jeden adres do komunikacji ze specjalnym rejestrem oraz grupę adresów do komunikacji z oddalonymi urządzeniami zewnętrznymi. Przystawka składa się z układu informacyjnego wysyłającego kod instrukcji czytaj (2), układu kontroli transmisji (1), układu nadawania (3) wysyłającego adres urządzenia i sygnał "CZYTAJ I", oraz wspólnego dla wszystkich urządzeń zewnętrznego rejestru, do którego wpisywana jest informacja sygnałem "CZYTAJ II"; rejestr jest zaopatrzony w swój numer wejścia/wyjścia.

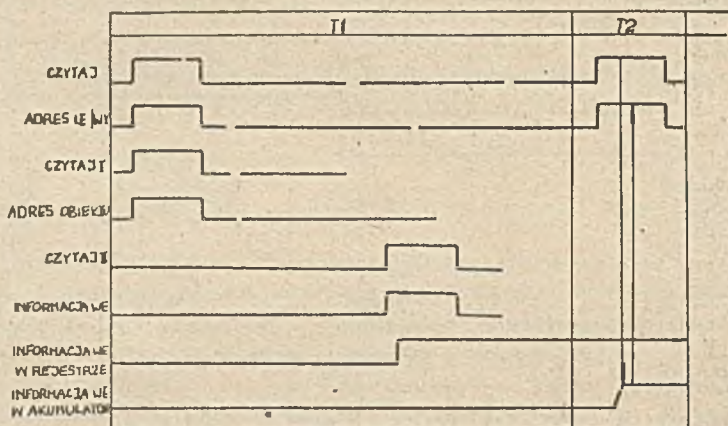
Warunkiem poprawności działania opisanego systemu jest dwukrotne, następujące po sobie wysłanie rozkazów CZYTAJ - pierwszego z numerem urządzenia, z którego chcemy przeczytać informację i drugiego z numerem rejestru 4.

Schemat blokowy układu pokazany jest na rys. 3, zaś przebiegi elektryczne na rys. 4.

Przebieg pracy układu jest następujący. Przed wysłaniem rozkazu czytaj należy przesłać do układu informacyjnego (2) (może on pracować pod tym samym numerem wejścia/wyjścia co rejestr zewnętrzny (4)) kodu instrukcji "czytaj". Następnie komputer wysyła kolejne dwa rozkazy "czytaj". Jeżeli pierwszy rozkaz "czytaj" jest adresem urządzenia z grupy oddalonych, układ kontroli transmisji i



Rys. 3.



Rys. 4.

spowoduje wysłanie z układu wysyłającego(2) sygnału "CZYTAJ" kodu instrukcji czytaj oraz adresu urządzenia. Po nadejściu żądana informacja jest wpisana sygnałem "CZYTAJ II" do rejestru zewnętrznego, skąd drugi rozkaz "CZYTAJ" przeczyta ją do akumulatora.

Uwagi

Podany sposób czytania informacji odbywa się przy ograniczonym oddaleniu urządzeń zewnętrznych. W podanych przykładach - w zależności od rodzaju przewodów - maksymalne oddalenie wynosi 500 - 700 m. W większości systemów sterowania jest to odległość w zupełności wystarczająca. W wypadku budowanej przystawki, czas trwania cyklu czytania informacji trwa dwa razy dłużej od cyklu komputerowego, jednak w zasadzie jest to i tak szybsza procedura niż przy stosowaniu transmisji asynchronicznej.

Istotną zaletą jest selektywne wybieranie informacji, umożliwione przez przesyłany kod rozkazu "czytaj", co z jednej strony przyspiesza cykl czytania informacji, a jednocześnie pozwala grupować zespół urządzeń pod jednym adresem wejścia/wyjścia.

Sama możliwość czytania informacji z urządzeń oddalonych ma sens, jeżeli także rozwiązany jest problem transmisji od komputera w kierunku "pisz". Jednak nie nastręcza on trudności, gdyż do tego celu można wykorzystać układ nadawania kodu instrukcji "czytaj", zaś potwierdzeniem przyjęcia słowa może być sygnał "CZYTAJ II". Przy zastosowaniu rozwiązania z mikroprocesorem należy prze-

strzegąc warunków, nie bezpośrednio po sobie nie mogą występować renkany "PISE" i "CZYTAJ", jednak w normalnej praktyce programowej zazwyczaj to nie zachodzi.

Zwiększając odległość urządzeń od komputera należy pamiętać o zastosowaniu specjalnych elementów logicznych (odpornych na zakłócenia na linii), służących do przesyłania informacji.

Przyjęty sposób adresowania urządzeń za pośrednictwem numeru komórki pamięci operacyjnej pozwala na podłączenie do komputera nieograniczonej praktycznie liczby urządzeń zewnętrznych, a więc rozwiązuje jeden z najbardziej istotnych problemów w systemach sterowania, tj. rozbudowy przez dołączanie nowych urządzeń bez konieczności zwiększania podzespołów elektronicznych w komputerze.

Przedstawiona organizacja przesyłania informacji preferuje stosowanie wspólnej szyny informacyjnej, przebiegającej od urządzenia do urządzenia. Nadajniki i odbiorniki informacji wszystkich urządzeń podłączone są do niej równolegle. W praktyce, szczególnie w dużych systemach, należy przewidywać prowadzenie kilku szyn przesyłania informacji połączonych przy komputerze w gwiazdę. Przy takim rozwiązaniu dołączanie nowego urządzenia sprowadza się do przedłużenia szyny informacyjnej.

mgr Elżbieta RÓSZKOWSKA

Instytut Maszyn Matematycznych
Zakład Doświadczalny

Dokumentowanie oprogramowania EMC

W pierwszym etapie opracowania oprogramowania (software) EMC należy ustalić jakiego typu dokumentacja (programowa i opisowa) jest niezbędna do wykonania tego zadania. Następnie należy opracować zasady przechowywania i zarządzania zbiorami danych różnego typu, związanymi z elementami wchodzącymi w skład oprogramowania. Element oprogramowania, zwany w dalszej części niniejszego artykułu pakietem, może być translatozem (języka Fortran, Cobol, PL/I) lub systemem (NARVIK).

Całość dokumentacji programowej i opisowej związanej z opracowywanym pakietem tworzy bibliotekę pakietu.

Dostatecznie wczesne rozwiązanie zagadnienia tworzenia i zarządzania biblioteką pakietu w znacznym stopniu ułatwia rozszerzanie i modyfikowanie elementów oprogramowania, co jest niezbędne dla jego rozwoju. Przykładem takiego podejścia do problemu produkcji software'u jest rozwój oprogramowania dla maszyn Jednolitego Systemu (RIAD). W pierwszym etapie pracy nad tym zagadnieniem ustalono zasady magazynowania i modyfikowania dokumentacji związanej z elementami tworzącymi system operacyjny OS/JS-P. Następnie zrealizowano środki programowe ułatwiające modyfikację i rozszerzanie bibliotek opracowywanych pakietów.

W niniejszym artykule przedstawiono koncepcję przechowywania biblioteki pakietu oraz sposób zrealizowania tej koncepcji w Zakładzie Doświadczalnym Instytutu Maszyn Matematycznych na przykładzie systemu operacyjnego OS/JS-P.

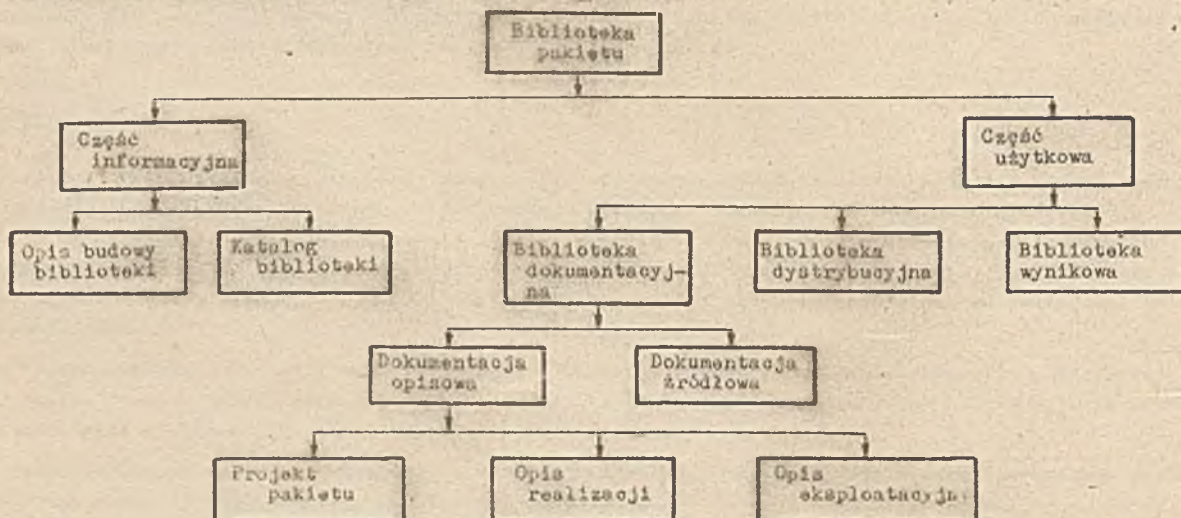
• Logiczna budowa biblioteki

Cała biblioteka pakietu składa się z dwóch części: informacyjnej i użytkowej.

Część informacyjna biblioteki pakietu wchodzącego w skład oprogramowania EMC zawiera informacje o budowie biblioteki pakietu oraz katalog pakietu tzn. spis elementów pakietu (modułów) wraz z informacjami, na jakich nośnikach one się znajdują.

Część użytkowa biblioteki pakietu wchodzącego w skład oprogramowania EMC dzieli się na biblioteki: • dokumentacyjną, • dystrybucyjną, • wynikową.

Schemat 1 przedstawia logiczną budowę biblioteki pakietu.



Schemat 1.

Biblioteka dokumentacyjna zawiera różnego typu dokumentację opisową oraz postać źródłową pakietu. Rodzaj przechowywanej dokumentacji opisowej zależy od typu pakietu. Na ogół będzie to dokumentacja związana z opracowaniem projektu pakietu, jego realizacją i eksploatacją.

Opracowanie projektu pakietu najczęściej składa się z trzech części:

- projekt wstępny, • wymagania dotyczące realizacji projektu, • projekt techniczny.

Dokumentacja opisowa związana z realizacją pakietu składa się z takich części jak:

- opis logicznej budowy pakietu oraz • szczegółowy schemat blokowy.

Dokumentacja opisowa związana z eksploatacją pakietu zawiera takie części jak:

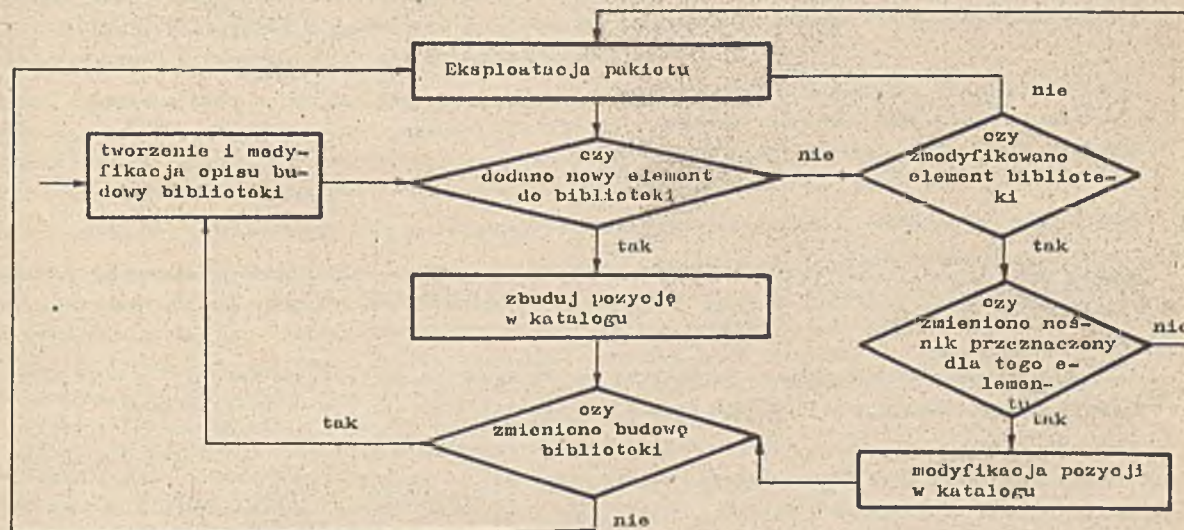
- instrukcja przejścia od postaci źródłowej do binarnej (dystrybucyjnej),
- instrukcja przejścia od postaci dystrybucyjnej do wynikowej, opis użytkowy pakietu.

Postać źródłowa pakietu jest to dokładny zapis działania danego pakietu przedstawiony w języku translatora dostępnego na danej EMC.

Biblioteka dystrybucyjna powstaje z postaci źródłowej pakietu jako wynik działania odpowiednich translatorów (jednego lub kilku). Biblioteka tego typu zawiera postać binarną całego pakietu i jest przeznaczona do przekazywania (dystrybucji) instalatorom pakietu w różnych ośrodkach.

Biblioteka wynikowa powstaje z biblioteki dystrybucyjnej po zastosowaniu przez instalatora specjalnego mechanizmu (generacja) pozwalającego na wybranie tylko tych fragmentów pakietu, które są potrzebne użytkownikom pracującym na danej instalacji EMC. W szczególnej sytuacji biblioteki dystrybucyjne i wynikowe mogą być identyczne.

Mechanizm modyfikowania biblioteki pakietu przedstawiają schematy: 2. - modyfikowanie części informacyjnej, 3. - modyfikowanie części użytkowej.



Schemat 2.

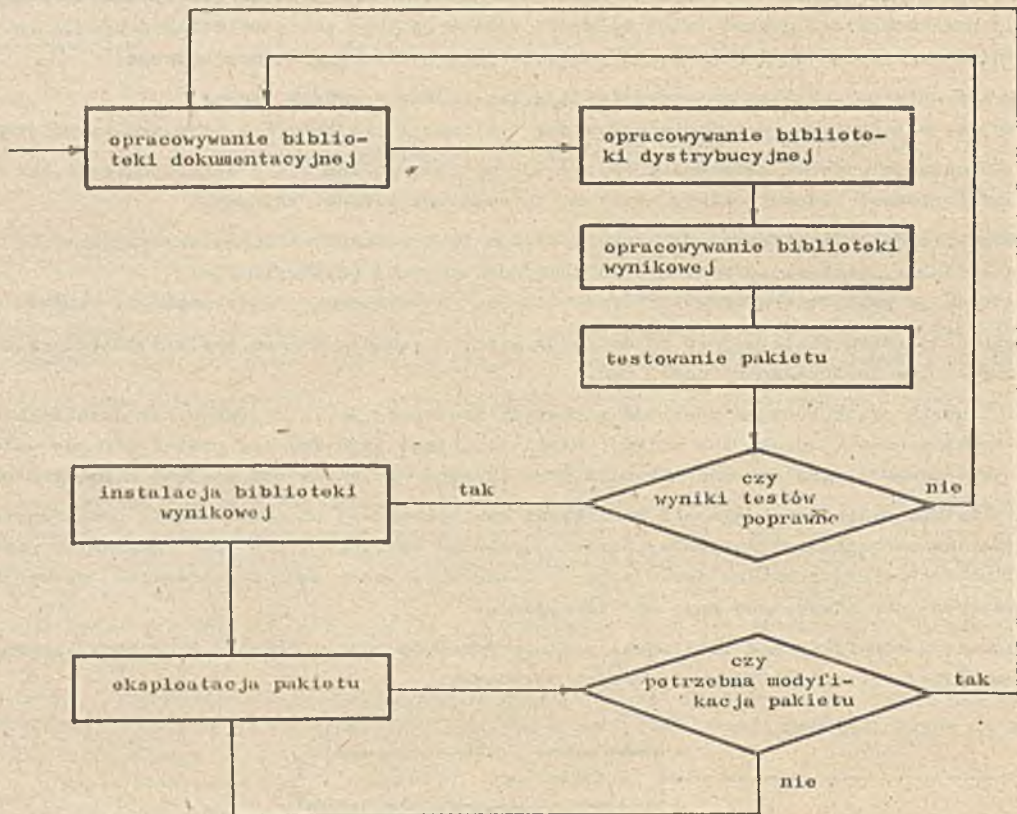
• Przechowywanie biblioteki pakietu

Zawartość biblioteki pakietu tworzą zbiory danych dwóch podstawowych typów:

- sekwencyjny
- biblioteczny - zawierający sekwencyjne człony, z których jeden, tj. katalog zbioru, umożliwia dostęp do pozostałych członów zbioru.

Nośnikami informacji, na których może być przechowywana biblioteka pakietu, a więc tworzące ją zbiory, są przede wszystkim pamięci zewnętrzne (pomocnicze) dołączone do EMC. Pamięci te można podzielić na dwa główne typy:

- pamięci o tylko sekwencyjnym dostępie do informacji (taśma magnetyczna),
- pamięci o bezpośrednim dostępie do informacji (dyski, bębny).



Schemat 3.

Szpula taśmy magnetycznej charakteryzuje się: dużą pojemnością, łatwością przenoszenia z jednej instalacji EMC na inną oraz tym, że mogą być umieszczone na niej jedynie zbiory typu sekwencyjnego.

Pamięci o bezpośrednim dostępie charakteryzują się szybszym dostępem do informacji oraz tym, że mogą zawierać zbiory danych różnego typu, w szczególności sekwencyjne i biblioteczne.

Wymienione tutaj cechy podstawowych pamięci zewnętrznych oraz wymagania nakładane na budowę biblioteki dystrybucyjnej czy wynikowej (często muszą być to zbiory typu bibliotecznego) wyznaczają typ nośnika zawierającego bibliotekę pakietu. Ze względu na łatwość przenoszenia z jednej instalacji EMC na inną (i dużą pojemność) podstawowym nośnikiem informacji służącym do przechowywania i dystrybucji biblioteki pakietu jest taśma magnetyczna. Natomiast opracowywanie biblioteki pakietu powinno odbywać się przy wykorzystaniu następujących nośników:

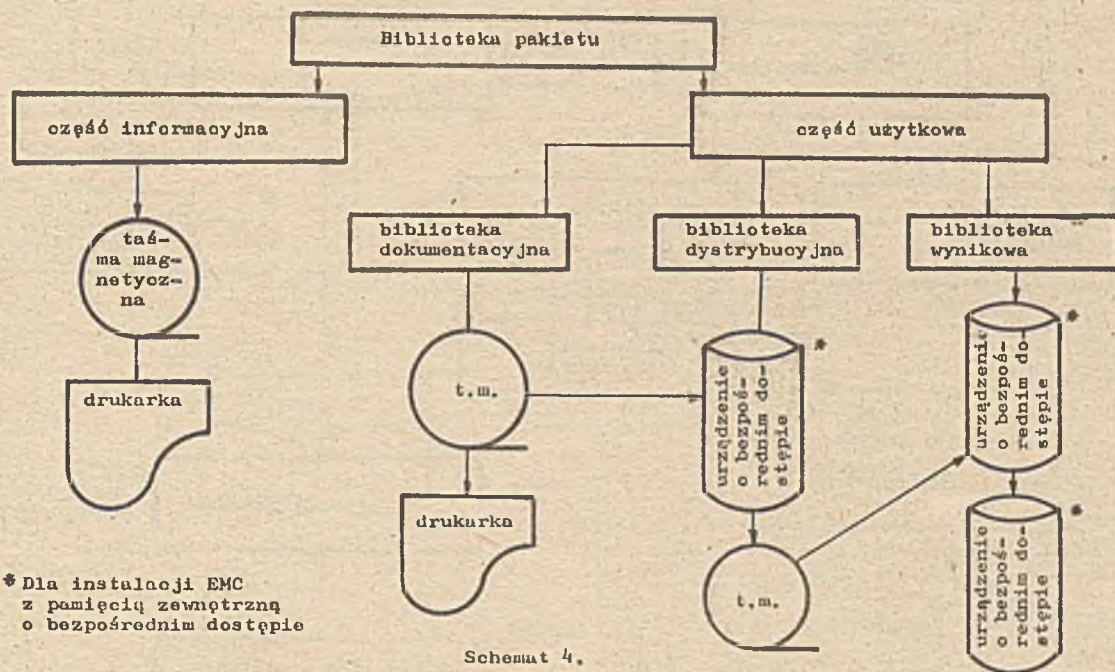
- taśma magnetyczna dla części informacyjnej biblioteki pakietu oraz biblioteki dokumentacyjnej,
- urządzenie o bezpośrednim dostępie dla bibliotek dystrybucyjnej i wynikowej.

Schemat 4. przedstawia rozmieszczenie biblioteki pakietu na maszynowych nośnikach informacji.

• Opracowywanie biblioteki pakietu

Pakiet przeznaczony do eksploatacji na EMC danego typu zwanej dalej wzorcową opracowywany jest na pewnej konkretnej EMC, którą nazwiemy technologiczną.

Jeżeli maszyny wzorcowa i technologiczna są kompatybilne wówczas z tej samej biblioteki dystrybucyjnej danego pakietu może powstać biblioteka wynikowa dla każdej instalacji EMC kompatybilnej



z technologiczną i wzorcową. Jeżeli maszyny wzorcowa i technologiczna nie są kompatybilne, to z biblioteki dystrybucyjnej danego pakietu może powstać biblioteka wynikowa dla każdej instalacji EMC kompatybilnej z wzorcową (ale nie z technologiczną).

Każdą konkretną EMC, na której dany pakiet będzie eksploatowany nazwiemy EMC eksploatacyjną. Maszyna eksploatacyjna powinna być zawsze kompatybilna z maszyną wzorcową. Gdy maszyny wzorcowa i technologiczna są kompatybilne to poszczególne składowe biblioteki pakietu opracowywane są przy wykorzystaniu środków maszynowych i programowych dostępnych na maszynie technologicznej.

Cała biblioteka pakietu może być przechowywana na nośnikach informacji maszyny technologicznej.

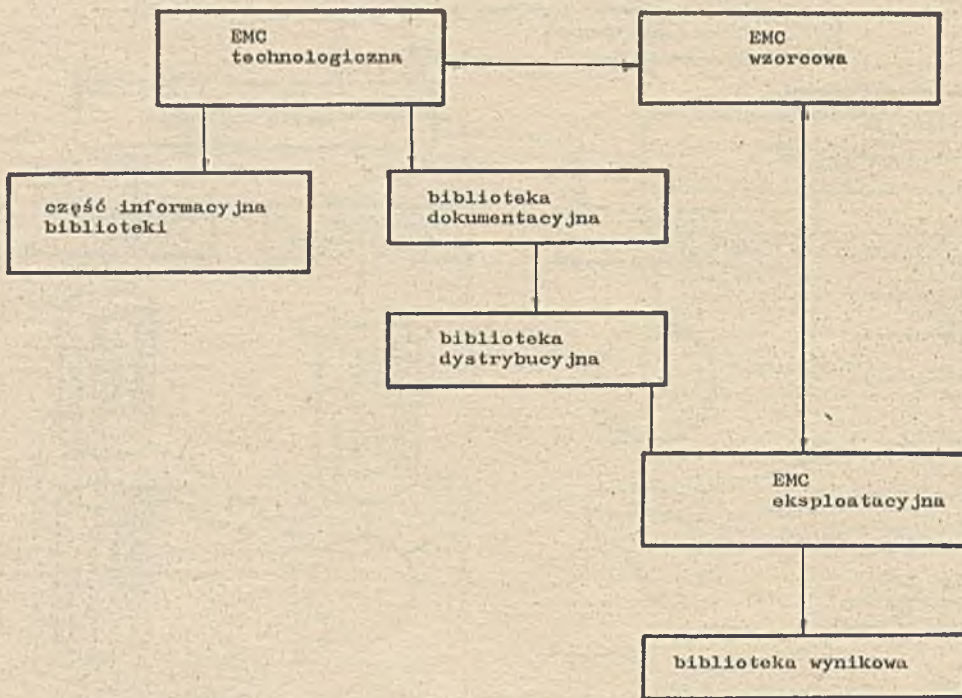
Maszyny wzorcowa i technologiczna mogą nie być kompatybilne, ale w tym wypadku konieczne jest przyjęcie założenia, że obie EMC (technologiczna i wzorcowa) mają co najmniej jedno urządzenie wejściowo-wyjściowe (tzn. takie, które ma możliwość wczytywania danych i wyprowadzania wyników) tego samego typu. Przy przyjęciu takiego założenia, możliwe jest opracowanie i przechowywanie na nośnikach maszyny technologicznej części informacyjnej biblioteki pakietu oraz biblioteki dokumentacyjnej. Biblioteki dystrybucyjna i przykładowa wynikowa (służąca do testowania pakietu) muszą być opracowywane i przechowywane na maszynie wzorcowej.

Przyjmijmy dodatkowo jedno z dwóch podanych poniżej założeń. Na maszynie technologicznej istnieje taki translator, który produkuje postać binarną pakietu zgodną z wymaganiami maszyny wzorcowej. Na maszynie technologicznej istnieje translator takiego języka, w którym potrafimy zapisać parametry sprzętu i oprogramowania maszyny wzorcowej oraz postać źródłową opracowywanego pakietu. Wówczas na maszynie technologicznej możemy opracowywać i przechowywać następujące składowe biblioteki pakietu:

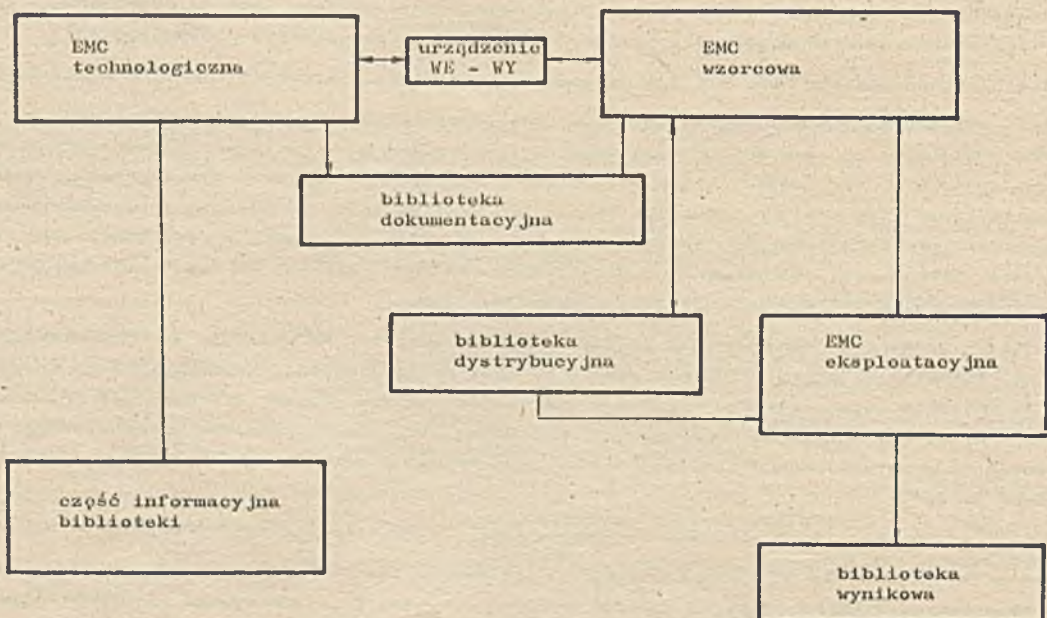
- część informacyjną, • bibliotekę dokumentacyjną, • bibliotekę dystrybucyjną.

Przykładową bibliotekę wynikową (służącą do testowania pakietu) opracowujemy na maszynie wzorcowej. Właściwa biblioteka wynikowa powinna być opracowywana i przechowywana na odpowiednich nośnikach maszyny eksploatacyjnej.

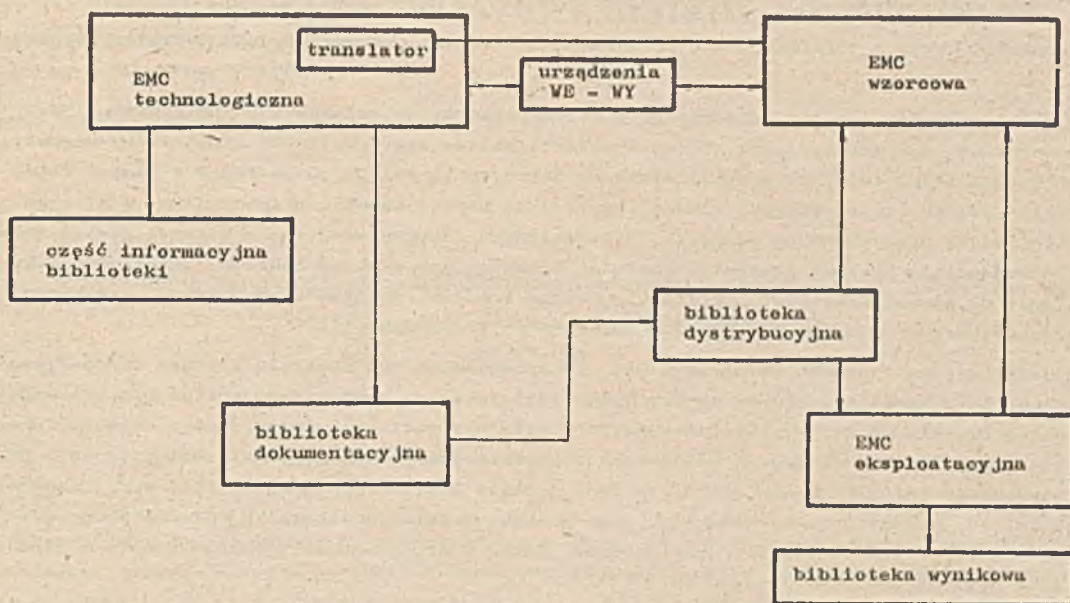
Poniższe schematy przedstawiają rozmieszczenie biblioteki pakietu na EMC: gdy maszyny wzorcowa i technologiczna są kompatybilne (5) oraz gdy maszyny wzorcowa i technologiczna nie są kompatybilne (6),(7).



Schemat 5.



Schemat 6.



Schemat 7.

• Biblioteka pakietu w systemie OS/JS

Dowolny pakiet pracujący pod kontrolą systemu OS/JS-P działający na EMC Jednolitego Systemu (RIAD) składa się z jednostek programowych takich jak: programy, podprogramy, jednostki tekstowe (makrorozkazy). Postać źródłowa każdej z wymienionych powyżej jednostek programowych wchodzi w skład dokumentacji źródłowej, a jej opis w skład dokumentacji opisowej. Cała dokumentacja źródłowa i opisowa tworzy bibliotekę dokumentacyjną pakietu.

Na podstawie informacji zawartych w bibliotece dokumentacyjnej produkowane są kolejne moduły biblioteki dystrybucyjnej. Przejście od postaci dystrybucyjnej do wynikowej odbywa się w sposób uzależniony od typu elementu oprogramowania. Składowe biblioteki pakietu wchodzącego w skład oprogramowania maszyn JS umieszczone są w pamięci o bezpośrednim dostępie jako zbiory typu bibliotecznego lub na taśmowym woluminie dokumentacyjnym. Każdy wolumin dokumentacyjny składa się z ciągu plików dokumentacyjnych stanowiących oddzielne pliki taśmowe. Początek i koniec pliku dokumentacyjnego wyznaczony jest przez znacznik początku i końca pliku. Pliki zorganizowane są w mniejsze jednostki logiczne zwane stronami. Początek i koniec strony wyznaczają również specjalne znaczniki. Strona składa się z 80-znakowych dokumentów.

Narzędziem przeznaczonym do tworzenia i aktualizacji woluminu dokumentacyjnego jest procesor POPR. Sterowanie pracą tego procesora odbywa się za pomocą specjalnego języka (POPR).

Opis fizycznej budowy woluminu dokumentacyjnego oraz języka POPR znajduje się w opracowaniach: "Budowa woluminów dokumentacyjnych", "Opis procesora języka aktualizacji woluminów dokumentacyjnych POPR".

Wszystkie dane tworzące informacyjną część biblioteki pakietu umieszczone są na rotacji woluminów informacyjnych zbudowanych zgodnie ze standardem POPR. Każda z rotacji woluminów informacyjnych składa się z czterech szpul taśmy magnetycznej. Trzy z tych taśm - taśmy rotacyjne, służą do dokonywania kolejnych poprawek w ramach rotacji, natomiast czwarta (taśma informacyjna) zawiera kopię aktualnej wersji woluminu informacyjnego. Taśmowy wolumin informacyjny na odpowiednich stronach kolejnych plików zawiera:

- opis logicznej budowy woluminów dokumentacyjnych wchodzących w skład biblioteki dokumentacyjnej pakietu opracowywanego pod kontrolą systemu OS/JS-P,

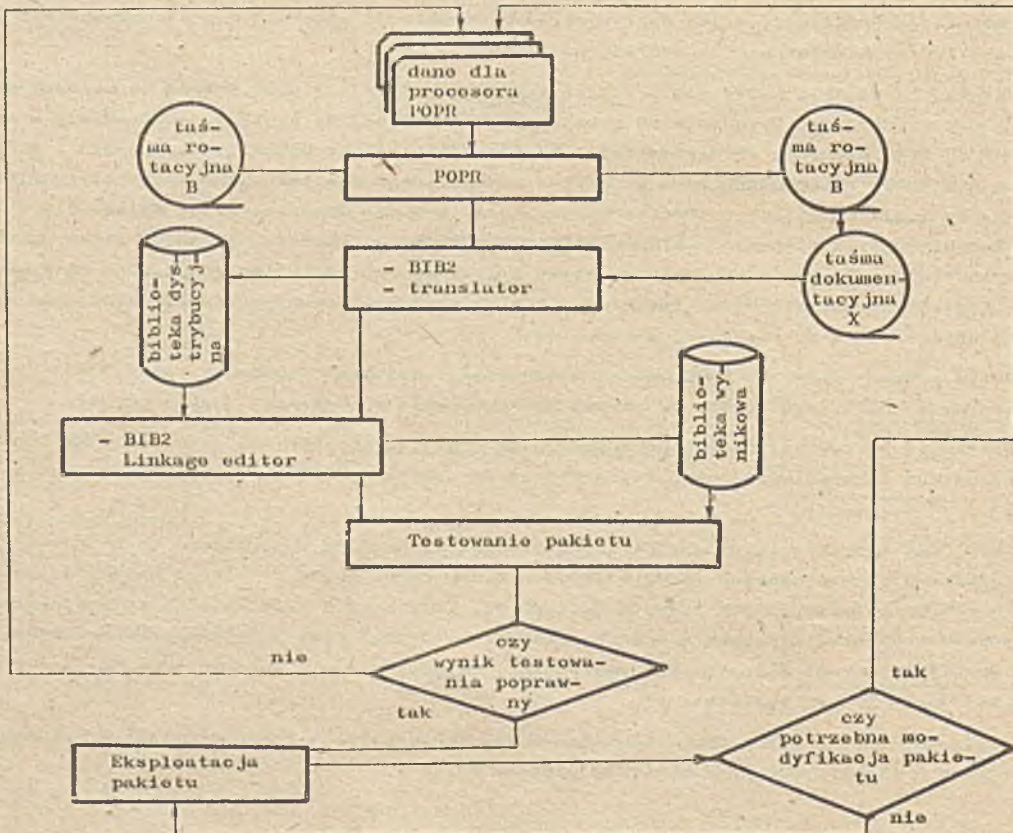
o spis zawartości (wraz z informacjami o nośnikach) biblioteki dokumentacyjnej i dystrybucyjnej pakietu.

Biblioteka dokumentacyjna pakietu wchodzącego w skład oprogramowania maszyn JS umieszczona jest na rotacji woluminów dokumentacyjnych. Każda z rotacji składa się z czterech szpul taśmy magnetycznej. Trzy z tych taśm (taśmy rotacyjne) służą do dokonywania kolejnych poprawek w ramach rotacji, natomiast czwarta (taśma dokumentacyjna) zawiera tę wersję materiału źródłowego, z którego utworzono bibliotekę dystrybucyjną pakietu. Pliki woluminu dokumentacyjnego zawierają postać źródłową oraz dokumentację opisową elementów (modułów) wchodzących w skład pakietu. Zawartość pliku podzielona jest na strony źródłowe, zawierające postać źródłową pakietu oraz opisowe, zawierające informacje dotyczące materiału źródłowego umieszczonego w danym pliku.

Postać dystrybucyjną elementu oprogramowania, opracowywanego pod kontrolą systemu OS/JS-P, tworzą na odpowiednim woluminie dyskowym zbiory typu biblioteczne zawierające postać binarną wszystkich elementów pakietu (modułów). Postać binarna powstaje z postaci źródłowej na podstawie informacji zawartych na stronach opisowych biblioteki dokumentacyjnej. Dla uzyskania automatyzacji tej czynności wspomniane informacje muszą być zapisane zgodnie z wymaganiami specjalnie opracowanego procesora BIB2. Wspomniany procesor traktuje inne (wskazane przez użytkownika) elementy oprogramowania (translatory) jako swoje podprogramy. Opis języka sterującego działaniem procesora BIB2 zawarty jest w opracowaniu "Opis procesora BIB2".

Biblioteka wynikowa pakietu opracowywanego pod kontrolą systemu OS/JS-P. powstaje z biblioteki dystrybucyjnej jako wynik działania innego elementu oprogramowania (linkage editor) lub na etapie generacji eksploatacyjnej wersji systemu OS/JS-P. Biblioteka tego typu zostaje umieszczona w zbiorach (najczęściej typu biblioteczne) na odpowiednim woluminie dyskowym, wskazanym przez instalatora. Zakłada się jednocześnie możliwość wyboru tych elementów pakietu, które są potrzebne w danej instalacji OS/JS-P.

Schemat 8. przedstawia metodę opracowywania biblioteki pakietu stosowaną w systemie OS/JS-P.



mgr inż. Zenon MITAL

mgr inż. Stanisław SZEJKO

Instytut Informatyki
Politechniki Gdańskiej

Zagadnienia automatyzacji badań silników spalinowych za pomocą dwuprocesorowego zestawu minikomputerów serii MERA-300

Wstęp

Dynamiczny rozwój przemysłu motoryzacyjnego rodzi liczne problemy, których rozwiązanie wymaga stosowania nowych metod i narzędzi badawczych. Dotyczy to m.in. zagadnień optymalizacji konstrukcji i zasad eksploatacji silników spalinowych w świetle różnych kryteriów, głównie zmniejszenia szkodliwego wpływu na środowisko oraz poprawy innych wskaźników ekonomiczno-eksploatacyjnych. Poza tym duża skala produkcji wymaga nowych, wydajnych narzędzi testowania gotowych wyrobów (dokonywanego metodą tzw. prób produkcyjnych).

Jedną z form badań silnikowych jest prowadzenie prób silników na stacjonarnych stanowiskach badawczych, tzw. hamowniach. Stanowią one zespół urządzeń umożliwiających zamocowanie badanego silnika oraz są wyposażone we wszystkie niezbędne instalacje zasilające (paliwowa, smarowania, chłodzenia, elektryczna), wentylacyjne a także odbiorniki energii (hamulce). Realizacja próby polega na zdjęciu odpowiednich charakterystyk silnika.

Informacje o obiekcie, instalacjach i zewnętrznym środowisku eksperymentu na stanowisku badawczym są uzyskiwane za pośrednictwem zbioru czujników pomiarowych. Oddziaływanie odbywa się za pośrednictwem zespołu regulatorów.

Wobec współczesnych wymagań dotyczących organizacji i zakresu badań, dokładności i wiarygodności uzyskiwanych danych oraz formy interpretacji wyników, tylko systemy komputerowe mogą być efektywnym narzędziem realizacji stawianych zadań. W Instytucie Informatyki Politechniki Gdańskiej zaprojektowano i zrealizowano system automatyzacji badań silników wysokoprężnych, zaimplementowany na minikomputerach serii MERA-300 i wdrożony oraz eksploatowany w OBRSK w Mielcu.

Szczegółowe omówienie tego systemu nie jest przedmiotem niniejszego artykułu. Wyczerpujący opis dostępny jest w literaturze wyszczególnionej na końcu opracowania. Tutaj natomiast chcemy podać pewne ogólniejsze uwagi dotyczące sposobów wykorzystania sprzętu minikomputerowego w systemach automatyzacji prób silników. Doświadczenia zebrane podczas realizacji i eksploatacji stworzonego systemu stały się bowiem podstawą do ogólniejszych rozważań na temat zakresu realizowanych zadań, istniejących ograniczeń oraz potrzeb stawianych w ramach problematyki automatyzacji badań silników spalinowych za pomocą sprzętu komputerowego, w szczególności z serii MERA-300. Jednym z efektów tych rozważań jest prezentowana w niniejszym artykule koncepcja wykorzystania zestawu dwuprocesorowego w systemie automatyzacji badań silników tłokowych, z zastosowaniem skonstruowanej w Instytucie Informatyki Politechniki Gdańskiej jednostki sterującej łączącej procesory Mowik 8b/100.

Potrzeby i ograniczenia występujące w zakresie automatyzacji prób silników spalinowych opartej na systemach minikomputerowych

W ramach automatyzacji prób silników spalinowych można wyróżnić kilka poziomów zadań, a mianowicie:

- w klasie zadań zbierania i przetwarzania danych
 - pomiary
 - nadzór nad bezpieczeństwem pracy obiektu
 - rejestracja wyników próby
 - przetwarzanie wyników próby
- w klasie zadań sterowania
 - sterowanie obiektem i instalacjami
 - sekwencyjne sterowanie przebiegiem próby
 - zarządzanie pracą zespołu hamowni

W zależności od zakresu dokonywanej automatyzacji badań pewne podzbiory zadań występujące w wyżej wymienionych poziomach realizowane są w systemie automatycznie.

Wśród pojawiających się na przestrzeni ostatnich lat różnych tendencji rozwoju komputerowych systemów automatyzacji prób silników spalinowych, początkowo dominowały tendencje do wykorzystywania dużych, uniwersalnych maszyn cyfrowych. Zaangażowanie dużych mocy obliczeniowych mogło być uzasadnione (w aspekcie ich efektywnego wykorzystania), tylko przy automatyzacji dużych hamowni, o kilkudziesięciu stanowiskach. Mankamentem takiego rozwiązania było globalne unieruchomienie badań w wypadku awarii komputera centralnego. Pojawienie się na rynku minikomputerów umożliwia tworzenie systemów o mniejszym stopniu centralizacji, często obsługujących pojedyncze stanowisko badawcze. Jednak w pewnych sytuacjach, przy zastosowaniu minikomputera w systemie automatyzacji badań silników spalinowych, pojawia się trudny do spełnienia warunek, wymagający scalenia dwu następujących elementów:

- prowadzenia złożonego eksperymentu badawczego w czasie rzeczywistym
- stworzenia środków programowania eksperymentu, realizacji algorytmów przetwarzania i różnorodnej interpretacji uzyskiwanych wyników.

Często analiza stawianych wymagań, w konfrontacji z charakterystykami dostępnego sprzętu minikomputerowego, uniemożliwia realizację wszystkich wymaganych funkcji w jednoprocessorowym, spójnym systemie. Szczególnie krytyczne znaczenie osiąga ten problem gdy zastosowany minikomputer ma małą moc obliczeniową. Wykorzystanie minikomputera ze stronicowaną pamięcią i ubogim repertuarem rozkazów maszynowych prowadzi do pamięciochłonnego programowania i wobec zakresu wymaganych od systemu funkcji zmusza do stosowania dodatkowych pamięci zewnętrznych. Wszystko to w sumie obniża efektywność czasową systemu.

Wymienione wyżej okoliczności, w obliczu narzucanych silnych uwarunkowań czasowych, uzasadniają spotykane niejednokrotnie w praktyce wypadki, kiedy niemożliwe jest stworzenie minikomputerowego systemu automatyzacji (spełniającego wszystkie stawiane wymagania) opartego na jednym procesorze. W tej sytuacji, w zależności od przeznaczenia systemu (a co za tym idzie - stawianych wymagań) i po uwzględnieniu możliwości dostępnego sprzętu minikomputerowego, mogą pojawić się dwa podejścia projektowe, wyznaczające generalną strukturę tworzonego systemu:

- rozdzielenie żądanych funkcji na dwa podsystemy
 - on-line, gdzie realizuje się te elementy wymagań, które muszą być spełniane podczas trwania procesu pomiarowo-sterującego
 - off-line, gdzie spełniane są funkcje, które nie muszą się realizować podczas trwania procesu pomiarowo-sterującego
- tworzenie systemu wieloprocessorowego.

Pierwszy z ww sposobów zastosowano w systemie automatyzacji prób silników wysokoprężnych, zrealizowanym w Instytucie Informatyki Politechniki Gdańskiej za pomocą minikomputerów serii MERA-300 [1],[2],[5],[6],[8], dokonano rozdziału wymaganych funkcji na podsystemy:

- stanowiskowy - bezpośrednie prowadzenie eksperymentu
- off-line - programowanie eksperymentu, przetwarzanie i interpretacja danych wynikowych.

W odniesieniu do prób produkcyjnych oraz innych badań normatywnych, dla których zrealizowany system jest przeznaczony, przedstawiony rozdział funkcji był możliwy do przeprowadzenia. Wynika to

z takiej struktury standardowych procedur badań silnikowych, która

- nie wymaga konieczności szybkiego uzyskiwania obszernej, pośredniej informacji podczas trwania procesu pomiarowo-sterującego, (dla uzyskania której niezbędna jest realizacja złożonych, czasochłonnych obliczeń), a od której uzależniony byłby dalszy przebieg eksperymentu
- umożliwia wcześniejszą i niezależną realizację większości zadań z zakresu programowania eksperymentu.

Osiągniętą funkcjonalność tak zorganizowanego systemu użytkownik zaakceptował. Ze względów organizacyjnych nawet korzystne okazało się wydzielenie tych wszystkich funkcji, które nie muszą być realizowane podczas trwania procesu pomiarowo-sterującego i zrealizowanie ich w trybie off-line na odrębnym sprzęcie, gdzie indziej zlokalizowanym i umożliwiającym obsługę potrzeb kilku (zautomatyzowanych lub nie) stanowisk hamowni.

Inaczej rzecz się ma z systemami o charakterze badawczym, przeznaczonymi do automatyzacji prób rozwojowych silników spalinowych. Tu często tok procedury badawczej jest mocno uzależniony od dotychczasowego przebiegu eksperymentu. W takich sytuacjach większość zadań z zakresu programowania przebiegu próby oraz przetwarzania i interpretacji wyników musi być realizowana w czasie rzeczywistym, umożliwiając prowadzącemu eksperyment uzyskiwanie bieżącej, wszechstronnej informacji wynikowej.

Odrębnym zagadnieniem zasługującym na uwagę jest realizacja pętli bezpośredniego sterowania cyfrowego (DDC). Obok niewątpliwych zalet, jakie daje stosowanie sterowania DDC (elastyczność algorytmów sterowania, niskie koszty sprzętu automatyki) należy też odnotować pewne mankamenty, wśród których dominuje poważne obciążenie mocy obliczeniowej komputera podczas trwania procesu pomiarowo-sterującego.

Konieczność realizacji wyżej opisanych wielu współbieżnych zadań w czasie trwania procesu pomiarowo-sterującego, w kontekście występujących silnych uwarunkowań czasowych, skłania do tworzenia systemu wieloprocesorowego.

Sprzętowe środki tworzenia systemów wielomaszynowych z wykorzystaniem minikomputerów serii MERA-300

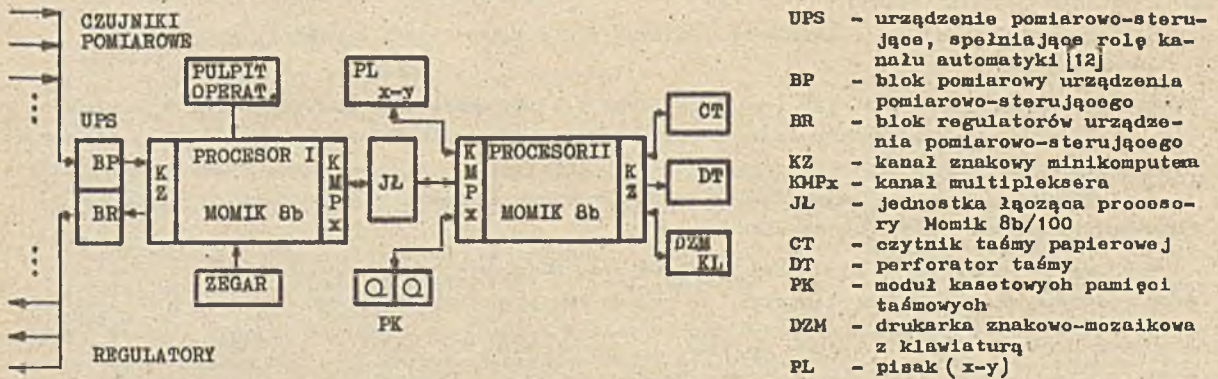
W Instytucie Informatyki Politechniki Gdańskiej stworzono sprzętowe możliwości łączenia minikomputerów serii MERA-300 w zestawy wielomaszynowe. Opracowano i zrealizowano jednostkę, łączącą przez kanały multipleksora dwa procesory Momik 8b/100. Szczegółowy opis tej jednostki zawarto w pracy [3]. Tu przytoczymy jedynie ogólną charakterystykę tego urządzenia, podając realizowane przez nie funkcje.

Zbiór rozkazów wykonywanych w jednostce łączącej umożliwia zainicjowanie i obsługę transmisji dowolnych obszarów pamięci między procesorami, a także dostarcza środków synchronizacji czasowej procesów obliczeniowych w obydwu procesorach. Transmisja odbywa się w wybranych podkanałach kanału multipleksora. Procesor A, chcąc przesłać blok informacji do procesora B, wysyła rozkaz PISZ do jednostki łączącej, co powoduje generację przerwania (zapytanie czytania) do procesora B. Rozkaz CZYTAJ, wysłany następnie z procesora B, inicjuje transmisję. Transmisja trwa do momentu pojawienia się sygnału końca transmisji z któregoś z procesorów. Towarzyszy temu generacja przerwania (koniec transmisji) do obydwu procesorów. Dodatkowo rozkaz PISZ PRZERWANIE umożliwia wysyłanie przerwania do drugiego procesora, dla zapewnienia synchronizacji programów wykonywanych w procesorach. Ponadto dla realizacji innych funkcji pomocniczych służą rozkazy DOŁĄCZ, ODŁĄCZ i PODŁĄCZ.

Możliwości wykorzystania zestawu dwuprocesorowego MERA-300 w systemie automatyzacji badań silników tłokowych

Na podstawie możliwości, jakie stwarza jednostka łącząca, rysuje się koncepcja konstrukcji dwuprocesorowego systemu automatyzacji badań silników tłokowych. System ten może być rozpatrywany nie tylko jako środek umożliwiający sprostanie stawianym potrzebom, ale również może być rozważany w aspekcie zwiększenia niezawodności i funkcjonalności.

W dalszej części niniejszego artykułu przedstawimy propozycję konfiguracji sprzętowej (rys.1) dwuprocessorowego systemu automatyzacji badań silników tłokowych, obsługującego hamownię jednostanowiskową. Podamy też sposób rozdzielania zadań spośród różnych poziomów automatyzacji na poszczególne elementy systemu oraz scharakteryzujemy oprogramowanie.



Rys.1. Konfiguracja sprzętu w dwuprocessorowym systemie automatyzacji badań silników tłokowych

Generalnie, procesor I realizuje zadania pomiarów, sterowania obiektem i instalacjami zasilającymi oraz sprawuje nadzór nad bezpieczeństwem pracy obiektu.

Procesor II steruje sekwencyjnym wykonywaniem próby, dokonuje przetwarzania danych oraz interpretacji i rejestracji wyników. Przewiduje się również, w razie awarii procesora II, realizację na procesorze I (w zredukowanej formie) pewnych zadań z zakresu sekwencyjnego sterowania próbą i rejestracji danych. Chroni to przed unieruchomieniem całego stanowiska prób w wypadku awarii procesora II, umożliwiając realizację podstawowych procedur badawczych. Przeprowadzana wówczas na procesorze I rejestracja danych umożliwia ich późniejsze wykorzystanie w trybie off-line.

System operacyjny czasu rzeczywistego dla przedstawionego zestawu sprzętowego, obejmujący oprogramowanie implementowane na obydwu procesorach, można pod względem funkcjonalnym podzielić na cztery główne składniki:

- grupa I a) jądro oprogramowania procesora I
b) programy wykonawcze procesora I
- grupa II a) jądro oprogramowania procesora II
b) programy wykonawcze procesora II.

Dla zapewnienia spójności poszczególnych elementów oprogramowania z obydwu grup, przewidziano w systemie operacyjnym mechanizmy synchronizacji czasowej i wymiany informacji między obydwu procesorami. W związku z tym w jądrze oprogramowania procesora I, obok programów obsługi przerw zewnętrznych (z obiektu, instalacji zasilających, pulpitu operatorskiego, HP, BR, zegara) i ekstrakodów (współpracy z kanałem automatyki, operacji arytmetycznych, maskowania przerw, przesyłania obszarów PAO w obrębie procesora I itd.), przewidziano (realizowane również ekstrakodowo) programy obsługi transmisji danych między procesorami I i II oraz programy obsługi przerw z procesora II dla umożliwienia synchronizacji procesów obliczeniowych w obu procesorach. Podobne środki, umożliwiające dwukierunkową komunikację między procesorami przewidziano również w ramach jądra oprogramowania procesora II, która obok tych funkcji spełnia ponadto obsługę standardowych urządzeń peryferyjnych (czytnik taśmy papierowej, drukarka DZM z klawiaturą, perforator taśmy papierowej, taśmowa pamięć kasetowa) oraz pisaka (x-y).

Poziom programów wykonawczych koduje odpowiednie algorytmy realizowane w różnych etapach trwania procedury badawczej silnika.

Zo względu na szeroki zakres zadań, realizowanych przez procesor II, jego praca odbywa się

w sposób nakładkowy i oparta jest na oprogramowaniu bibliotecznym zawartym na taśmowej pamięci kasetowej. Oprogramowanie to obejmuje w ramach obsługi funkcji wcześniej wyszczególnionych m.in. takie elementy, jak obszerne pakiety programów korekcyjnych wg różnych międzynarodowych norm budai silników spalinowych, oraz programy aproksymacji i kształlenia charakterystyk silnika.

Uwagi końcowe

Zastosowanie dwuprocessorowego zestawu minikomputerów serii MERA-300 do konstrukcji systemu prób silników spalinowych, pozwala na badania o szerokich możliwościach funkcjonalnych, znacznie rozszerzających zakres automatyzacji tych badań. Na uwagę zasługuje fakt, że tworzenie zestawu dwuprocessorowego nie wymaga żadnych zmian konstrukcyjnych w samych minikomputerach.

Literatura

- [1] DEHNEL J., GÓRSKI J., MITAL Z.: Komputerowy system automatyzacji stanowisk pomiarowych silników wysokoprężnych. Wrocław: PWrocl. 1976 Prace Naukowe Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej, z.41
- [2] MITAL Z., RAATZ M., SZEJKO S.: Komputerowy system automatyzacji badań silników wysokoprężnych. Sympozjum nt "Automatyzacja eksperymentalnych badań maszyn", Warszawa 1978
- [3] KOWALEWSKI L., KUCHCIŃSKI K., SZEJKO S., WISZNIEWSKI B.: Sposób i jednostka do łączenia minikomputerów z serii MERA-300 w zestawy wielomaszynowe. Gdańsk: PGdańska 1978
- [4] GÓRSKI J., ŁOŚ R., MITAL Z., SOLTYSIK J.: Możliwości zastosowań minikomputerów serii MERA-300 do automatyzacji przemysłowych procesów pomiarowo-sterujących. Konferencja naukowo-techniczna "Zastosowanie komputerów w przemyśle", Szczecin 1978
- [5] System operacyjny SASP dla minikomputera Momik 8b/100 przeznaczony do sterowania sekwencyjnego procesem pomiarowo-kontrolnym. Gdańsk: PGdańska 1976 Prace Instytutu Informatyki Politechniki Gdańskiej, z.5
- [6] MANKIEWICZ J., WIEREMJEWICZ J.: Komputerowe przetwarzanie danych pomiarowych w systemie automatycznego badania maszyn cieplnych. Sympozjum nt. "Automatyzacja eksperymentalnych badań maszyn", Warszawa 1978
- [7] DZIURLA B., RAATZ M., SZEJKO S.: Język sekwencyjnego sterowania eksperymentem STER. Wrocław: PWrocl. 1976 Prace Naukowe Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej, z.41
- [8] SZEJKO S., WIEREMJEWICZ J.: Komputerowy system graficznej interpretacji danych. Konferencja naukowo-techniczna "Zastosowanie komputerów w przemyśle", Szczecin 1978
- [9] GÓRSKI J., ŁOŚ R., SOLTYSIK J.: Structured hierarchical real-time operating system for a sequential process control. IFAC/IFIP Workshop on Real-Time Programming, Marienhann, Finland, June 19-21, 1978
- [10] ŁOŚ R., SOLTYSIK J., SCHREIBER J.: Projekt modularnego oprogramowania systemu pomiarowo-sterującego dla badań maszyn cieplnych w czasie rzeczywistym. Wrocław: PWrocl. 1976 Prace Naukowe Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej, z.41
- [11] Minikomputerowy system sterowania próbami produkcyjnymi turbin gazowych. Cz.I - System operacyjny sterowania sekwencyjnego w czasie rzeczywistym dla minikomputerów serii MERA-300. Gdańsk: PGdańska 1977 Prace Instytutu Informatyki Politechniki Gdańskiej, z.7
- [12] CICHY M., MAZUREK S.: System informatyczny do automatyzacji badań maszyn cieplnych SIMAC. Praca Instytutu Informatyki Politechniki Gdańskiej, z.1
- [13] ŁOŚ R., SOLTYSIK J.: Metoda strukturalnego projektowania komputerowych systemów sterowania próbami produkcyjnymi maszyn cieplnych. Gdańsk: PGdańska 1978

mgr inż. Jerzy MANKIEWICZ
mgr inż. Stanisław SZEJKO
mgr inż. Jan WIEREMJEWICZ

Instytut Informatyki
Politechniki Gdańskiej

Wielodostępny system symulacji minikomputerów MOMIK 8b/100

Nauczanie podstaw organizacji maszyn cyfrowych i programowania w językach niższych rzędów, a także obsługi operatorskiej emc, stanowi jeden z najbardziej kłopotliwych elementów dydaktyki kierunków informatycznych. Wiadomości zdobyte w ramach tych zajęć stanowią podstawę dla licznych przedmiotów: od pracowni sprzętowych i oprogramowaniowych poprzez organizację systemów i techniki programowania do projektowania systemów sterowania i przetwarzania danych. W tej sytuacji zasadniczym problemem staje się organizacja laboratorium spełniającego dwa warunki:

- wyposażenie w sprzęt analogiczny do wykorzystywanego w dalszych latach nauki;
- zapewnienia możliwości równoczesnej pracy (nauki) dużej liczby osób bezpośrednio przy maszynach.

Uwzględnienie dodatkowo - zwykle niewystarczającej ilości sprzętu (koszt) powoduje konieczność ciągłego kompromisu między liczebnością grup studenckich pracujących przy emc a przydzielanym im czasem pracy.

Bliższe przyjrzenie się tradycyjnym metodom nauczania tego zakresu materiałowego pozwala stwierdzić, iż:

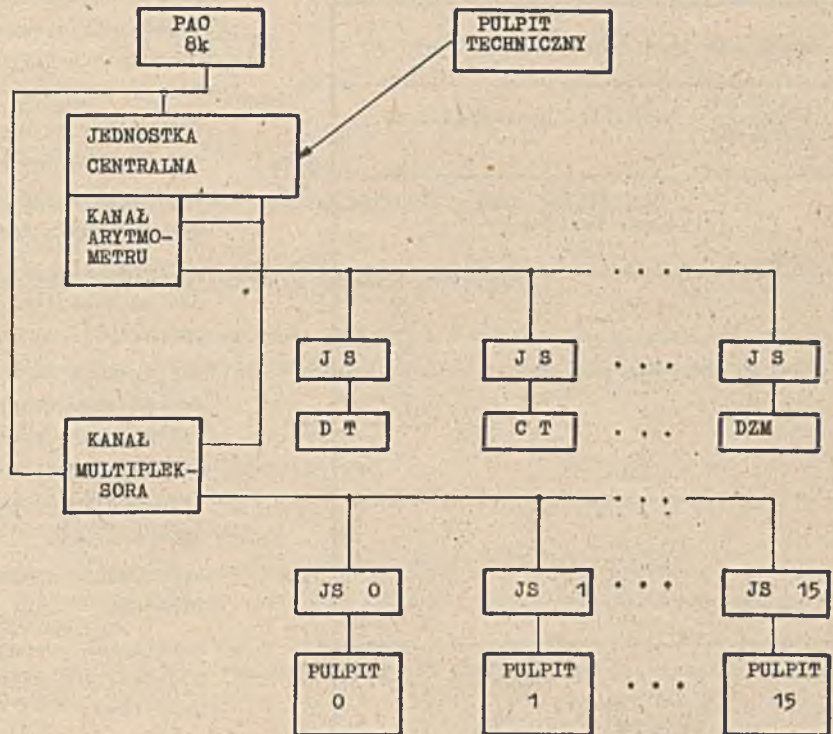
- możliwości obliczeniowe laboratoryjnych emc wykorzystywane są zwykle w bardzo niewielkim stopniu,
- programy są proste i w małym stopniu wykorzystują obszary pamięci operacyjnej,
- znakomitą większość czasu zajmuje nie wykonywanie się lecz wprowadzanie i uruchamianie programów, często z pulpitu technicznego.

Taka ocena sytuacji doprowadziła - w Instytucie Informatyki Politechniki Gdańskiej do:

- wyposażenia laboratorium podstaw programowania i organizacji maszyn w popularny i stosunkowo tani sprzęt minikomputerowy serii MERA 300; nie bez znaczenia była tu możliwość kontynuacji nauczania w kierunku prac hardware'owych i projektowo-konstrukcyjnych;
- stworzenia systemu umożliwiającego równocześnie naukę większym grupom osób; cel ten osiągnięto przez dołączenie pewnej liczby standardowych pulpity operatorskich do minikomputera MERA 303, na którym zasymulowano rzeczywistą maszynę dla każdego z dołączonych pulpity; całkowita zgodność pulpity i zasymulowanych maszyn z faktyczną organizacją procesora MOMIK 8b/100 daje - przy pracy wielodostępnej - pełne złudzenie korzystania z rzeczywistego minikomputera; prezentacja struktury i działania tego systemu a także wstępna ocena przydatności stanowią zasadniczą treść niniejszego artykułu;
- kontynuacji rozbudowy laboratorium w systemy analogiczne, odwzorowujące inne organizacje maszyn cyfrowych - w trakcie realizacji jest zespół pulpity i wielodostępny system symulacji maszyny ODRA 1305.

Konfiguracja sprzętu wykorzystywanego przez omawiany wielodostępny system symulacji przedstawiona jest na rys. 1. Wykonane w Instytucie Informatyki Politechniki Gdańskiej pulpity stanowią rejestry kluczy i lampek dołączone jako adresowane urządzenia wejścia/wyjścia. Jeden pulpit zajmuje jeden podkanał kanału multipleksera a zastosowana jednostka sterująca umożliwia sygnalizację zmiany stanu kluczy pulpitu, dokonywanie transmisji do/od i może znaleźć zastosowanie w organizacji współpracy z praktycznie dowolnym typem pulpitu.

W celu zapewnienia możliwości równoczesnej pracy wszystkim użytkownikom stworzono programowy symulator minikomputera MOMIK 8b, który w sposób interpretacyjny wykonuje funkcje - cyklicznie



Rys. 1. Konfiguracja sprzętu tworzącego wielodostępny system symulacji minikomputerów MOMIK 8b

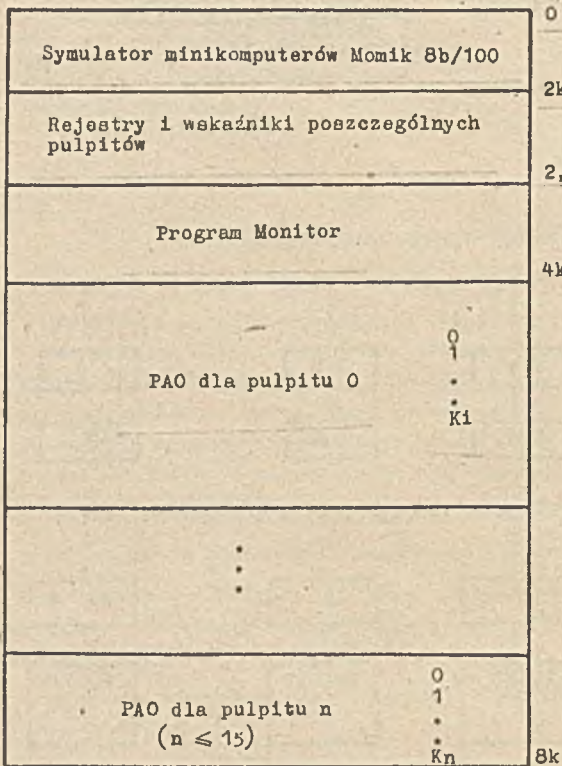
dla wszystkich pulpitu. Wymagało to ponadto wyodrębnienia dla każdego z pulpitu obszaru pamięci operacyjnej odzwierciedlającej jego symulowane rejestry, wskaźniki i pamięć operacyjną. Mapa PAO, wraz z szacunkową zajętością poszczególnych modułów, przedstawiona jest na rys. 2.

Symulator reaguje na zmiany kluczy na dołączonych pulpituach wykonując żądane operacje na symulowanych rejestrach i pamięci oraz przysyłając odpowiednie informacje na lampki pulpitu. W skład symulatora wchodzi następujące moduły programowe:

- moduł sterujący obsługą cykliczną, zapewniający wielodostępną pracę maksymalnie 16 pulpitu,
- moduł organizacji transmisji w kanale multipleksera, tj. odczytanie stanu kluczy i wysyłanie informacji do rejestrów lampki,
- moduł reakcji na zmiany stanu kluczy różnego typu oraz
- moduł symulacji działania jednostki centralnej zgodnie ze stanem kluczy pulpitu.

Ponieważ system zapewnia cykliczną organizację pamięci przypisaną każdemu z pulpitu oraz istnienie wyróżnionych komórek PAO strony zerowej i pierwszej, użytkownik korzystający z pulpitu ma wrażenie pracy na odrębnym minikomputerze o zmniejszonej pamięci i wydłużonym czasie wykonywania rozkazu. System wykonuje 800-1200 rozkazów symulowanych w ciągu sekundy przydzielając cyklicznie czas potrzebny na wykonanie jednego rozkazu pracującym pulpitu. Oznacza to, że, w zależności od ilości wykonujących się programów, maksymalny czas reakcji systemu dla każdego pulpitu wynosi od 1 do 16 ms.

Korzystanie z urządzeń wejścia/wyjścia pracujących w kanale arytmometru odbywa się na normalnych zasadach z dodatkiem - ze względu na ograniczoną ich liczbę - automatycznej rezerwacji urządzeń. Możliwa jest także przerwanowa praca urządzeń dołączonych w klasie wejścia/wyjścia, przy czym system zapewnia wpisywanie przerw do symulowanych rejestrów przyjęć, oczywiście tylko tego pulpitu, dla którego dokonano rezerwacji urządzenia.



Ki - pojemność pamięci dla i-tego pulpitu
($i=1,2,...n$)

Rys.2. Mapa pamięci systemu

W celu zwiększenia efektywności systemu, zapewnienia elastyczności i wygody obsługi system wyposażono w program Monitor, którego główne funkcje obejmują:

- konwersacyjną współpracę z prowadzącym zajęcie (operatorem systemu),
- przydział i ochronę pamięci operacyjnej dla dołączonych pulpitów,
- sterowanie przydziałem urządzeń zewnętrznych dla poszczególnych użytkowników (rezerwaacja nadrzędna),
- wczytywanie taśm binarnych programów do obszarów symulowanych PAO poszczególnych pulpitów; wczytywanie pracuje przerwanowo i zapewnia protokół pamięci,
- wyprowadzanie dowolnych obszarów symulowanych pamięci,
- wypisywanie zawartości tych obszarów na DZM-180,
- symulowanie przerw z możliwością "wstrzykiwania" ich do rejestrów przyjęć wybranych pulpitów,
- zliczanie niektórych błędów manualnych i programowych z podaniem ich ilości.

Ponadto system przystosowany jest także do symulacji systemu wieloprocesorowego opartego na wspólnej pamięci operacyjnej.

Prezentowany system wykorzystywany jest praktycznie w Zakładzie Przetwarzania Informacji Instytutu Informatyki Politechniki Gdańskiej od października 1978 r. do nauki podstaw organizacji i programowania maszyn cyfrowych (III rok specjalności Informatyka i Automatyka). We wdrożonej wersji praca wielodostępna odbywa się za pośrednictwem czterech pulpitów dołączonych do standardowej konfiguracji MERA 303.

Wstępna ocena wykorzystania systemu pozwala na sformułowanie następujących wniosków:

- nastąpiło znaczne zwiększenie dostępności sprzętu dla studentów, a co za tym idzie - wydłużenie czasu efektywnej nauki bezpośrednio przy maszynie,
- osiągnięto możliwość indywidualnego przydziału zadań studentom,
- zwiększono efektywność wykorzystania sprzętu minikomputerowego, także przez odciążenie innych zestawów,
- znacznie spadła awaryjność sprzętu wskutek odizolowania nie przeszkolonych jeszcze osób od rzeczywistej maszyny,
- system spowodował znaczne ułatwienie pracy prowadzącemu zajęcia,
- pojawiła się potrzeba dołączenia do systemu większej liczby standardowych urządzeń zewnętrznych: czytniki, perforatory, drukarki,
- potwierdzono przydatność systemu do nauki programowania w języku wewnętrznym i assemblerze, a przede wszystkim - organizacji maszyn, włącznie ze współpracą z urządzeniami zewnętrznymi i systemem przerw.

Literatura

- [1] MANKIEWICZ J., WIEREMJEWICZ J.: Opracowanie systemu pulpitów dołączonych do maszyny cyfrowej MOMIK 8b/100. Praca dyplomowa Instytutu Informatyki Politechniki Gdańskiej nr 2/1977
- [2] Firmowa dokumentacja techniczna zestawów minikomputerowych serii MERA-300

mgr inż. Lech KOWALEWSKI
mgr inż. Krzysztof KUCHOCIŃSKI
mgr inż. Bogdan WISZNIEWSKI

Instytut Informatyki
Politechniki Gdańskiej

System wielomaszynowy oparty na minikomputerach z serii MERA-300

Łączenie maszyn cyfrowych w zestawy wielomaszynowe: potrzeby i możliwości

Jedną z najbardziej dynamicznie rozwijających się w ostatnich latach dziedzin projektowania systemów liczących jest łączenie maszyn cyfrowych. Maszyny/procesory łączy się w zestawy w celu uzyskania:

- większej niezawodności systemu liczącego,
- większej mocy obliczeniowej i pojemności pamięci systemu,
- budowy dużych systemów możliwie małym kosztem (szczególnie przy łączeniu minikomputerów),
- możliwości sterowania obiektami rozproszonymi.

Oprócz powyższych bardziej ogólnych tendencji mogą jeszcze pojawić się inne przyczyny, związane ze szczególnym zastosowaniem budowanego systemu, np. przy łączeniu dużych maszyn bardzo odległych od siebie w sensie geograficznym.

W każdym systemie połączonych maszyn cyfrowych czy procesorów wyróżnia się dwa podstawowe elementy, z których jest on zbudowany:

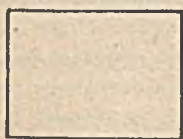
- element przetwarzający, którym może być zarówno maszyna cyfrowa (systemy wielomaszynowe) jak i sam procesor (systemy wieloprocessorowe),
- element łączący, zwany również ścieżką łączącą, którym może być zarówno specjalistyczna jednostka łącząca jak i łącze radiowo czy nawet gonio z plikiem kart perforowanych.

W niektórych systemach występuje również element przełączający (komutujący), którego funkcją jest wybór ścieżki łączącej. Może być to zarówno wyspecjalizowany element przetwarzający jak układ mechaniczny czy nawet urządzenie pocztowe. Rys. 1 przedstawia każdy z trzech wymienionych wyżej elementów.

ELEMENT
PRZETWARZAJĄCY

ELEMENT
PRZELĄCZAJĄCY

ŚCIEŻKA
ŁĄCZĄCA

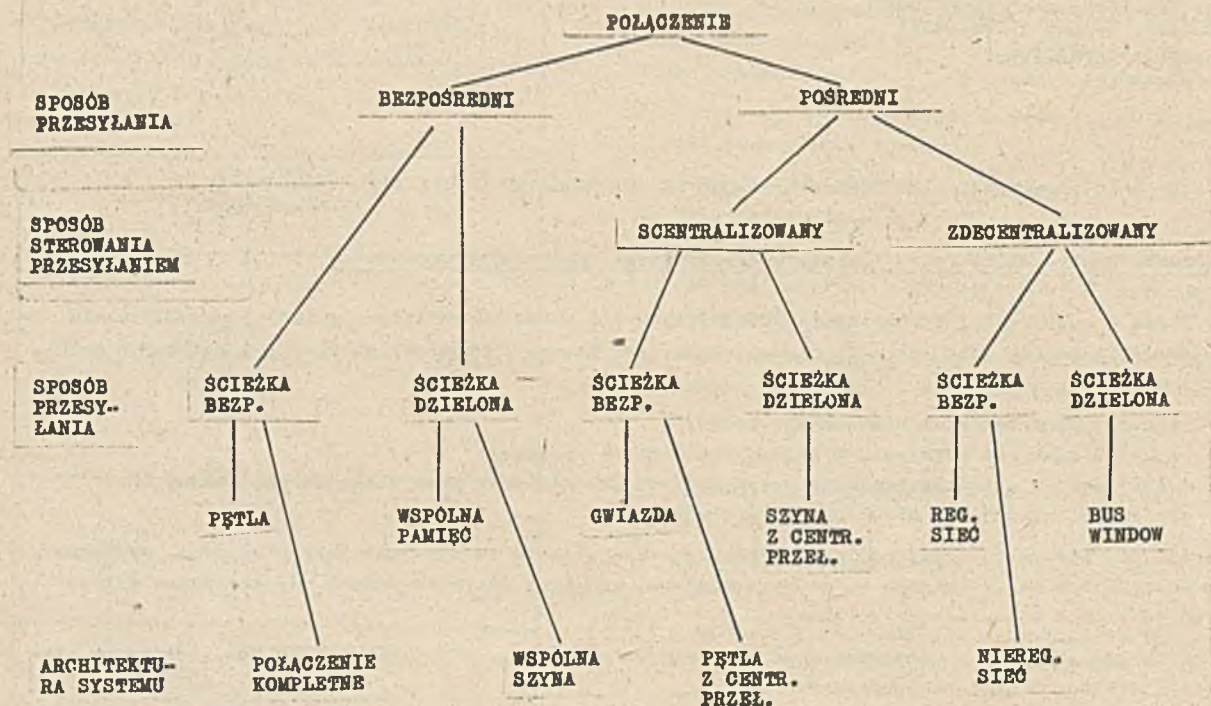


Określenie różnych konfiguracji ułatwia tzw. inżynierskie drzewo decyzyjne, szeregujące wszelkie możliwe zestawy wielu elementów przetwarzających (rys. 2), zaproponowane w pracy [1].

Pierwszym elementem opisanym przez drzewo decyzyjne jest "pętla" (ang. direct dedicated loop). Zestawy takie stosuje się tam, gdzie jest wymagana duża modularność systemu, a nie duża prędkość przesyłania informacji (np. w obrębie jednej uczelni, zakładu produkcyjnego czy laboratorium badawczego).

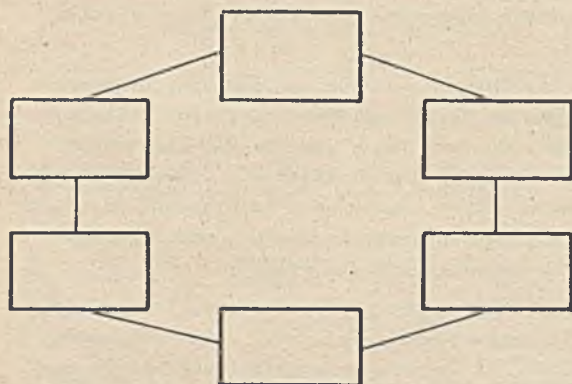
Rys. 1. Podstawowe elementy składowe systemów wielomaszynowych/processorowych

Układy typu "pętla" (rys. 3) powstają przy projektowaniu komunikacyjnych układów transmisji danych i składają się z pewnej liczby pojedynczych elementów przetwarzających, z których każdy połączony jest z dwoma sąsiadującymi elementami. Przesyłanie informacji w pętli może być dokonywane w obu kierunkach, lecz w praktyce ze względu na wiele prostsze oprogramowanie przepływ jej odbywa się tylko w jedną stronę. Każdy element łączący, po odbiorze informacji musi sprawdzić, czy jest ona przeznaczona dla niego. Jeśli nie, przesyła ją dalej. Informacja nie mająca odbiorcy krąży w systemie. Nadmierna jej ilość może zablokować system. Adresowanie informacji z reguły odbywa się przez adresowanie konkretnego procesu, który jest odbiorcą informacji, a nie danego



Rys. 2. Inżynierskie drzewo decyzyjne

procesora. Proces-odbiorca identyfikuje się ze swoim adresem i akceptuje przesłaną informację. Tego typu adresowanie jest niezależne od liczby pracujących elementów przetwarzających i jest stosowane także w wielu innych systemach. Najbardziej znanym przykładem takiego systemu jest Rozproszony System Komputerowy (Distributed Computer System) w Uniwersytecie w Irvine, Kalifornia. Składa się on z pięciu minikomputerów rozproszonych po miasteczku akademickim. Łączy szeregowo zapewniają przepustowość 2,3 Mbit/s i są zdublowane dla poprawienia niezawodności systemu. Istnieją też przełączniki omijające minikomputery w wypadku ich awarii.



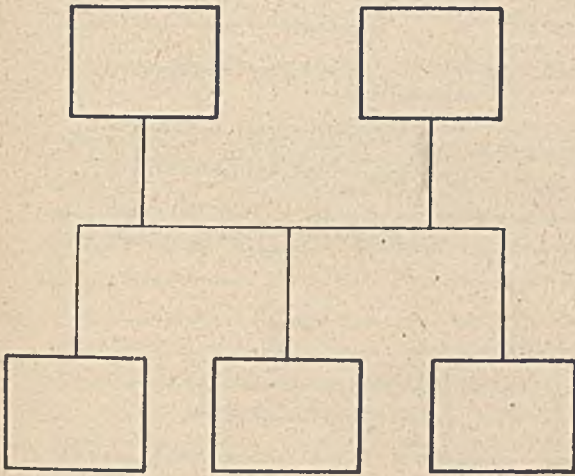
Rys. 3. Układ typu "Pętla"

W systemach pracujących w czasie rzeczywistym często stosuje się zestawy wielomaszynowe z "wspólną szyną" (ang. direct shared bus). System ten (rys. 4) wykorzystywany jest w przemyśle, lotnictwie, kosmonautyce i w urządzeniach wojskowych. Zaletą systemu jest duża możliwość natychmiastowej rekonfiguracji w razie awarii elementów przetwarzających. Ponadto prawie zawsze stosuje się zdublowanie szyny łączącej. Dla takiej konfiguracji elementy przetwarzające dzielą między sobą szynę wg określonego algorytmu alokacji i informacje przesyłane są bezpośrednio od nadawcy "na szynę" i do odbiorcy, który identyfikuje się z daną transmisją i akceptuje ją. Dołączenie dodatkowych elementów przetwarzających nie pociąga za sobą dodatkowych kosztów, zaś maksymalna liczba dołączonych elementów zależy od możliwości adresacji. Bardzo trudne jest zwiększenie przepustowości i prędkości przesyłania wspólnej szyny. Najczęściej w takiej sytuacji trzeba wymienić całą szynę. Jednym z

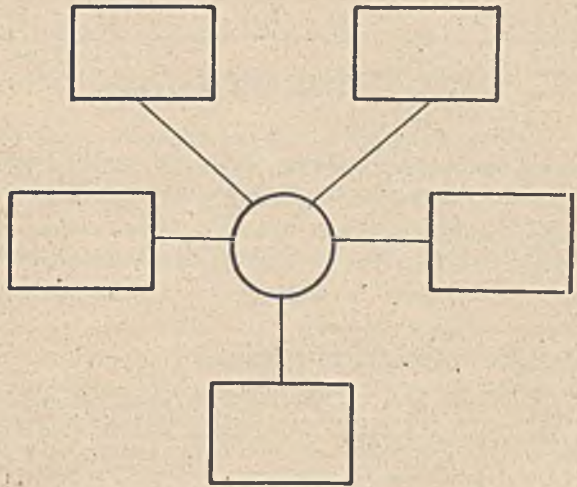
przykładów opisanego powyżej systemu jest wokółziemski system nawigacyjny Marynarki Wojennej Stanów Zjednoczonych.

Tam, gdzie sposób wymiany informacji jest bardziej złożony, ozy ostrzejsze są wymagania dotyczące jej ochrony, stosuje się układy z elementami przełączającymi rozbudowanymi do różnych rozmiarów i przystosowanymi do pełnienia różnych funkcji dodatkowych.

Jednym z rozwiązań jest system "gwiazda" (ang. indirect centralized dedicated star). System ten (rys. 5) składa się z centralnego elementu przełączającego i z pewnej liczby elementów przetwarzających, połączonych z nim dwukierunkowymi ścieżkami łączącymi. Informacje są przesyłane między elementami przetwarzającymi za pośrednictwem centralnego elementu przełączającego, który może również spełniać rolę bufora. Z punktu widzenia elementów przetwarzających element przełączający jest jedynym źródłem i odbiorcą informacji. Ścieżkę do odbiorcy wybiera przełącznik na podstawie informacji sterujących podanych przez nadawcę. Jednym z przykładów takiego systemu jest Network/440 firmy IBM, w którym pracują maszyny typu IBM 360 połączone łącznie dzierżawionymi z centralnym układem typu 360/91.



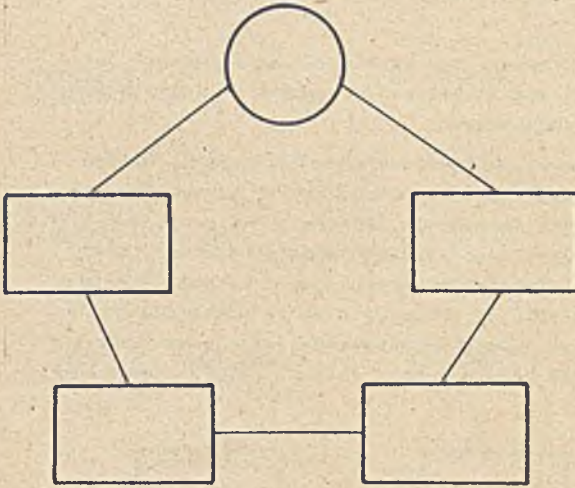
Rys. 4. Układ ze "wspólną szyną"



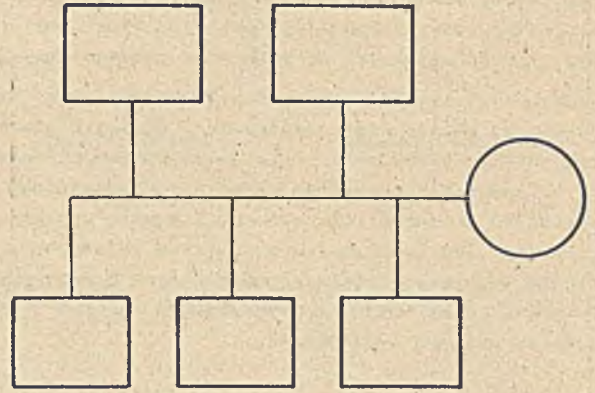
Rys. 5. Układ typu "gwiazda"

Innym rozwiązaniem jest "pętla z centralnym przełącznikiem" (ang. indirect centralized dedicated loop). Układ ten jest bardzo podobny do opisywanej już "pętli" (rys. 6). Różnica polega na tym, że aby informacja przeszła od nadawcy do odbiorcy musi przejść przez element przełączający, którego zadaniem jest odpowiednie przygotowanie informacji oraz jej właściwa adresacja. Rozbudowa systemów tego typu jest tańsza niż "gwiazdy", ponieważ nie trzeba prowadzić ścieżki łączącej aż do elementu przełączającego lecz tylko do najbliższych sąsiadów. Sterowanie komunikacją jest bardziej złożone od układów typu "gwiazda" ze względu na dodatkowe wymagania stawiane elementom przełączającym pracującym w "pętli" (jako przekazniki). Przykładem takiej konfiguracji jest eksperymentalny system przesyłania danych SPIDER, łączący jedenaście komputerów w laboratorium Bell'a w Murray Hill N.Y. w Stanach Zjednoczonych. Dane w tym systemie przesyłane są w zunifikowanych blokach z prędkością 1.5 Mbit/s. Funkcję centralnego elementu przełączającego spełnia mini-komputer.

Do układów z elementami przełączającymi należy również układ z "szyną z centralnym przełącznikiem" (ang. indirect centralized shared bus). Z funkcjonalnego punktu widzenia jest on identyczny z "gwiazdą" (rys. 7). Jedyna różnica polega na tym, że elementy przetwarzające podłączone są do elementu przełączającego przez wspólną szynę, a nie przez indywidualne ścieżki łączące. Nadawca chcący przesłać informację musi zgłosić potrzebę przydzielenia mu szyny, następnie przesłać informację do przełącznika, z którego następuje retransmisja poprzez tę samą szynę do odpowiedniego odbiorcy. Ciekawym przykładem takiego systemu jest system ALOHA. Poszczególne maszyny



Rys. 6. Pętla z centralnym przełącznikiem



Rys. 7. Układ typu "szyna z centralnym przełącznikiem"

połączone są "szyną" radiową. Jeden dwupłeski kanał radiowy w paśmie 24 kHz łączy centralny element przełączający w postaci jednej maszyny IBM 360/65 z odległymi końcówkami-minikomputerami.

Pozostałe typy połączeń opisanych w przedstawionym na wstępie drzewie decyzyjnym są rzadko wykorzystywane w praktyce ze względu na różne wady, zniechęcające konstruktorów. Są one jednak rozważane w literaturze fachowej; opisano je bliżej np. w pracy [1].

Realizacja systemu dwumaszynowego

Autorzy tego artykułu postanowili stworzyć działający system wielomaszynowy dla potrzeb Instytutu Informatyki Politechniki Gdańskiej. Założyliśmy przy tym, że system powinien:

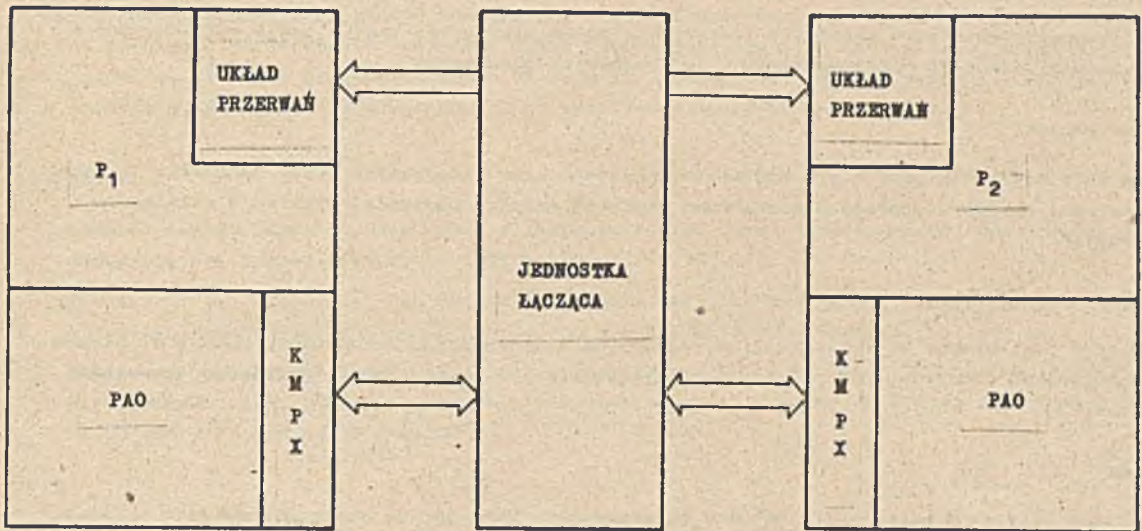
- być w najwyższym stopniu uniwersalny, tj. składać się wyłącznie z takich jednorodnych elementów, które łatwo można rozłączyć w celu stworzenia dowolnej konfiguracji,
- być zbudowany jak najmniejszym nakładem kosztów, tj. z wykorzystaniem posiadanego przez Instytut Informatyki PG sprzętu obliczeniowego.

Wybór padł na powszechnie używane w naszym Instytucie minikomputery z serii MERA-300, wyposażone w procesor MOMIK 8b/100, spełniający drugie nasze założenie.

Minikomputer ten jest małą maszyną cyfrową o pojemności PAO 8 kbyte, o stosunkowo bogatych możliwościach współpracy z urządzeniami zewnętrznymi przez kanały: arytmometru, bezpośredniego dostępu i multipleksera, umożliwiającego równoczesną współpracę w szesnastu podkanałach.

Pierwsze założenie spełnia zaprojektowana i wykonana przez nas jednostka łącząca, zapewniająca połączenie przewodowe minikomputerów z serii MERA-300 na odległość do 15 metrów. Zapewnia ona bezpośrednie połączenie między dwoma minikomputerami przez wybrany podkanał (jeden z szesnastu) kanałów multipleksersów w każdym z nich. Minikomputer współpracuje z jednostką łączącą, jak z każdym innym urządzeniem podłączonym do kanału multipleksora i wymiana informacji między maszynami odbywa się według ogólnie przyjętych dla tego kanału zasad (rys. 8).

Najprostszym sposobem przeprowadzenia transmisji z jednego minikomputera (P1) do drugiego (P2) jest zainicjowanie operacji kanałowej "PISZ". Jednostka łącząca generuje wówczas przerwanie "DAJ CZYTAJ" do P2, inicjującego w odpowiedzi operację kanałową "CZYTAJ", łączność zostaje nawiązana i trwa do chwili, gdy wyszeruje się licznik długości obrotu do transmisji. W tej samej chwili jednostka łącząca generuje przerwanie "KONIEC TRANSMISJI" do P1 i P2, informując je o jej zakończeniu. Oczywiście, jak w każdej transmisji blokowej przez kanał multipleksersa, w czasie jej trwania każdy z minikomputerów niezależnie i równocześnie wykonuje swoje własne programy.



Rys. 8. Połączenie dwóch maszyn z serii MERA-300 za pomocą jednostki łączącej

Jednostka łącząca zapewnia również możliwość generowania innych przerwań (do czterech), tj. takich które mogą być wykorzystane do współpracy z innymi urządzeniami (np. automatyki przemysłowej) lub dodatkowych połączeń z P1/2/. Oprócz transmisji blokowych jednostka zapewnia realizację programowego odłączenia P1/2/ od P2/1/, ponownego przyłączenia, czytania słowa stanu jednostki przez P1 i P2. Sposób i jednostkę do łączenia minikomputerów z serii MERA-300 w zespoły wielomaszynowe zgłoszono do Urzędu Patentowego PRL [3].

System dwumaszynowy zbudowaliśmy na podstawie wykonanej wcześniej jednostki łączącej oraz dwóch minikomputerów z serii MERA-300. Stanowi on pierwszą praktyczną realizację koncepcji budowy uniwersalnego systemu wielomaszynowego. Po uruchomieniu zestawu i opracowaniu dla niego oprogramowania realizującego operacje kanałowe, stało się możliwe zrealizowanie przez nas interesującego zastosowania, tzn. systemu MINI-BASIC-77. Szczegółowy opis systemu MINI-BASIC-77 zawarto w pracy [2].

Budowa systemów wielomaszynowych. Zastosowanie

Jednostka łącząca ze względu na swoją uniwersalność zapewnia możliwość połączenia większej liczby minikomputerów MERA-300, tzn. jeden maksymalnie z szesnastoma innymi (tj. liczba podkanałów). Poszczególne minikomputery mogą ponadto współpracować z zewnętrznymi pamięciami masowymi (np. dyski, taśmy magnetyczne), co stwarza prawie nieograniczone możliwości rozbudowy systemów prawdziwie wielomaszynowych, o dużych pamięciach.

Dysponując tą uniwersalną jednostką i minikomputerem serii MERA-300 można zestawiać różne systemy wielomaszynowe o interesującej konfiguracji.

Spójrzmy jeszcze raz na zaprezentowane na wstępie decyzyjne drzewo inżynierskie (rys. 2) i prześledźmy jakie konfiguracje można zestawić opierając się na naszej jednostce i minikomputerach z serii MERA-300.

• Pętla

Liczba elementów w pętli jest teoretycznie nieograniczona. W praktyce może ją ograniczać jedynie złożoność realizowanych procesów i wymagania środowiska.

• Połączenia kompletne "każdy z każdym"

Jedynym ograniczeniem liczby elementów w tej strukturze jest liczba podkanałów, co znaczy, że

jeden minikomputer może mieć szesnastu towarzyszy, zatem sieć w tej konfiguracji może składać się maksymalnie z siedemnastu minikomputerów i 126 jednostek łączących.

• Wspólna pamięć

System taki mógłby składać się z piętnastu minikomputerów dołączonych przez jednostki łączące do szesnastego, współpracującego z zewnętrzną pamięcią masową, najlepiej dyskiem i pełniącego funkcję pamięci.

• Wspólna szyna

Układ taki jest trudny do realizacji ze względu na organizację minikomputera MERA-300, chociaż jest to możliwe po skonstruowaniu specjalnego urządzenia - wspólnej szyny, dołączonej do każdej z jednostek.

• Gwiazda

Jeśli jako element przełączający użyjemy minikomputera MERA-300 to otrzymamy gwiazdę składającą się maksymalnie z szesnastu minikomputerów dołączonych do minikomputera-przełącznika.

• Pętla z centralnym przełącznikiem

Przy użyciu jednego z minikomputerów jako elementu przełączającego w tej pętli, system ten różni się od zwykłej pętli oprogramowaniem.

• Szyna z centralnym przełącznikiem

Przy próbie realizacji tej konfiguracji pojawiają się podobne trudności, jak przy układzie ze wspólną szyną.

• Regularna sieć

Zasadą konstruowania takiej sieci jest połączenie każdego z minikomputerów z taką samą liczbą maszyn, co każdy pozostały w sieci oraz, że każdy z nich musi mieć zdolność komutowania informacji. Od struktury typu "połączenie kompletne - każdy z każdym" regularna sieć różni się właśnie tą ostatnią cechą. W szczególności dopuszcza się, aby minikomputery na krawędziach sieci miały mniejszą liczbę sąsiadów niż pozostałe (tj. wewnątrz sieci).

• Nieregularna sieć

Sieć nieregularna jest uogólnieniem sieci regularnej, różniące się zasadniczo od niej oprogramowaniem. Warunkiem koniecznym jest bowiem posiadanie przez każdy z elementów przetwarzających pełnej informacji o strukturze systemu, podczas gdy dla sieci regularnej taki warunek nie jest wymagany. W konfiguracji tej wykorzystuje się wyłącznie niektóre, już istniejące (stąd nieregularność) maszyny cyfrowe. Uniemożliwia to budowanie sieci nieregularnych opartych na minikomputerach, szczególnie z serii MERA-300.

• Okno

Połączenie takie jest możliwe wyłącznie dla minikomputerów o strukturze typu wspólna szyna i jego realizacja oparta na minikomputerze MERA-300 jest niemożliwa.

Zastosowanie

Możliwość budowania uniwersalnych struktur wielomaszynowych opartych na tym tanim, łatwo dostępnym minikomputerze i na skonstruowanym przez nas urządzeniu-jednostce łączącej otwiera zupełnie nowe możliwości zastosowań tego minikomputera, których wymaganiom nie byłby on w stanie sprostać w konfiguracji wyłącznie jednomaszynowej.

Gdy trzeba powiększyć pojemność pamięci czy szybkość przetwarzania systemu, skrócić czas reakcji czy zwiększyć niezawodność wystarczy tylko wybrać odpowiadającą naszym celom konfigurację i zrealizować ją własnymi środkami. Często okazuje się to znacznie tańsze, prostsze i równie sprawne w działaniu, co złożone systemy oparte na kosztownym sprzęcie importowanym.

Literatura

- [1] Andersen G.A., Jensen E.D.: Computer interconnection structures: taxonomy, characteristics and examples. Computing Surveys 1975 vol.7 nr 7 s.197-213
- [2] Kowalewski L., Kuchciński K., Szejko S., Wiszniewski B.: Translator języka BASIC w oparciu o system dwóch współpracujących procesorów z serii MERA-300. Informatyka 1979 nr 2
- [3] Kowalewski L., Kuchciński K., Szejko S., Wiszniewski B.: Sposób i jednostka do łączenia mikokomputerów serii MERA-300 w zespoły wielomaszynowe, zgłoszony do Urzędu Patentowego PRL dn. 11.08.1978 pod nr P. 209002

Biuletyn Informacyjny OBIEKTOWE SYSTEMY KOMPUTEROWE

mgr inż. Lech KOWALEWSKI
mgr inż. Krzysztof KUCHCIŃSKI
mgr inż. Bogdan WISZNIEWSKI

Instytut Informatyki
Politechniki Gdańskiej

Niektóre problemy rekonfiguracji systemów wielomaszynowych zbudowanych na maszynach cyfrowych serii MERA-300

Wybrane aspekty niezawodnościowe systemów wielomaszynowych/procesorowych

Przy konstruowaniu systemów liczących prócz spełnienia podstawowych założeń, wynikających z ich przeznaczenia (jak opisano bliżej w pracy [2]), ważne jest zapewnienie dużej niezawodności.

Przez niezawodność jakiegokolwiek systemu rozumie się jego zdolność do wykonywania określonego zadania. Zadanie oznacza realizowanie określonych czynności w danych warunkach pracy przez ustalony okres czasu t . Matematyczną miarą niezawodności jest prawdopodobieństwo, że system spełni powyższe wymagania w przedziale czasu $(0, t)$.

Jedną z metod zapewniających wysoką niezawodność systemów liczących jest budowanie ich z wielu elementów przetwarzających, zatem w żadnym wypadku awaria jednego z nich nie może oznaczać awarii całego systemu. Wyróżnia się trzy rodzaje wymagań stawianych "niezawodnym" systemom wielomaszynowym/procesorowym.

• Wymaganie pracy ciągłej

Do tej kategorii systemów należą systemy sterowania procesami kontroli ruchu powietrznego i inne systemy o działaniu bezpośrednim. Miarą niezawodności R tej klasy systemów jest prawdopodobieństwo poprawnej pracy w zadanym przedziale czasu:

$$R = p(\Delta t)$$

gdzie $p(\Delta t)$ oznacza prawdopodobieństwo poprawnej pracy systemu w przedziale czasu Δt . W opisie niezawodnościowym takich systemów podaje się zwykle średni czas między uszkodzeniami - MTBF (ang. mean time between failures). Dla takich systemów - w szczególności sterujących procesami technologicznymi - MTBF wynosi wiele tysięcy godzin.

• Wymaganie krótkiego czasu trwania przestoju

Użytkownikami takich systemów są najczęściej placówki handlowe, biura podróży i wszyscy inni dopuszczający krótkie przerwy w pracy systemu. Nie toleruje się jednak zbyt długich przestoju.

Miarą niezawodności takich systemów jest prawdopodobieństwo, że po usunięciu awarii system będzie pracował poprawnie przez dany okres czasu:

$$R = G(t)p(T)$$

gdzie: $G(t)$ oznacza prawdopodobieństwo, że w chwili t uszkodzenie zostanie usunięte, $p(T)$ oznacza prawdopodobieństwo właściwej pracy systemu przez okres czasu T licząc od chwili t . W literaturze $G(t)$ nosi często nazwę współczynnika gotowości systemu [4]. Systemy takie opisuje się najczęściej średnim czasem naprawy - MTR (ang. mean time to repair).

• Wymaganie nienaruszalności bazy danych

Wymaganie to jest istotne dla systemów o potężnych bazach danych, gdzie zachowanie integralności tej bazy jest znacznie ważniejsze niż czas ponownego uruchomienia systemu. Miara niezawodności R systemów tej klasy jest współczynnik gotowości:

$$R = G(t)$$

W dalszej części pracy zajmujemy się systemami spełniającymi pierwsze z wymienionych wyżej wymagań, tj. zespołami wielomaszynowymi/procesorowymi o działaniu bezpośrednim i pracy ciągłej. Metodą realizacji tego wymagania jest rekonfiguracja systemu w trakcie jego pracy w wypadku detekcji awarii modułu. W wyniku rekonfiguracji system kontynuuje pracę na niezmiennym poziomie lub ogranicza zakres wykonywania funkcji. Każdą z tych właściwości systemu przyjęto nazywać w literaturze odpowiednio: bezpiecznym upadkiem i łagodnym upadkiem.

Oprócz podanych wyżej miar niezawodności systemów zwanych potocznie parametrami niezawodnościowymi, charakteryzującymi je właściwie tylko od strony technicznej, wprowadza się również pojęcie efektywności systemu. Przez efektywność systemu rozumie się stopień realizacji funkcji systemu. Innymi słowy jest to prawdopodobieństwo, że system o określonej niezawodności wykona wszystkie swoje zadania w danych warunkach eksploatacji, w przewidzianym przedziale czasu. Jest to parametr pozwalający porównywać różne organizacje systemów liczących, ich zdolność do polnienia określonych funkcji mimo uszkodzeń sprzętu. Miara ta jest bardziej przydatna do oceniania systemów niż same tylko prawdopodobieństwa ich poprawnej pracy, bowiem różne systemy zbudowane z takich samych, choćby najbardziej niezawodnych elementów przetwarzających i łączących mają różną efektywność ze względu na organizację systemu, tj. konfigurację sprzętu i jego oprogramowanie. Dla większości systemów, mających k stanów niezawodnościowych efektywność określa się jako:

$$E(t) = \sum_{i=0}^k e_i p(s_i, t)$$

gdzie: s_i oznacza i -ty stan pracy systemu,

$p(s_i, t)$ prawdopodobieństwo i -tego stanu systemu w chwili t ,

e_i wskaźnik efektywności i -tego stanu systemu, określany stosunkiem wyjściowych efektów działania systemu w stanie s_i do wyjściowych efektów, jakie byłyby uzyskane w wyniku działania całkowicie sprawnego systemu (w stanie s_0).

Przejęcie systemu z jednego stanu do drugiego następuje bądź w wyniku celowej zmiany jego konfiguracji, bądź też rekonfiguracji spowodowanej wystąpieniem uszkodzenia. W tym drugim wypadku dąży się do utrzymania możliwie stałej efektywności systemu. Szczegółowe rozważania na temat prawdopodobieństwa poprawnej pracy komputerów i systemów komputerowych zawarto np. w pracy [3].

W rekonfiguracji wyróżnia się następujące etapy:

- detekcja uszkodzenia,
- lokalizacja uszkodzenia,
- reorganizacja sieci połączeń (dobór grupy sprawnych modułów mogących realizować zadania),
- sprawdzenie poprawności stanu programu (danych, zbiorów itp.),
- przywrócenie wartości stanu programu, który uważany jest za wiarygodny,
- restart.

Etapy te realizuje się sprzętowo i programowo, rzadziej ręcznie. Sprzętowo ze wspomaganie programowym realizuje się na ogół trzy pierwsze etapy, a programowo ze wspomaganie sprzętowym pozostałe. Po rekonfiguracji wskazane jest testowanie systemu.

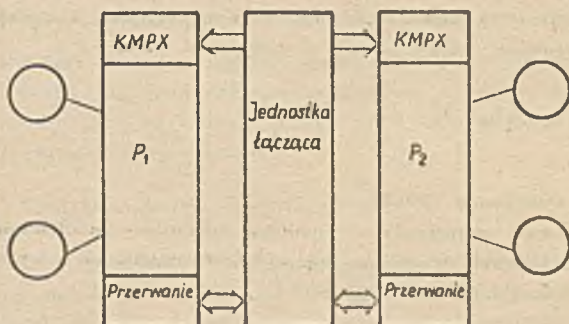
Powyższe etapy rekonfiguracji narzucają wymagania dotyczące budowy systemu. Wymagania dotyczące systemu operacyjnego, zarządzania zadaniami (ang. scheduling), kontroli błędów i restartu, można ująć w następujących punktach [1]:

- system operacyjny powinien być tak skonstruowany, aby mógł efektywnie wykorzystywać maksymalnie liczną konfigurację jak i dowolny podsystem;
- wszystkie tablice systemowe i zarządzające zadaniami formuje się w podwójnej ilości i umieszcza w różnych miejscach pamięci systemu (tj. w różnych modułach pamięci);
- ważniejsze bloki funkcjonalne powinny mieć kontrolę poprawnej pracy, która nie powinna wpływać na efektywność wykonywanych przez blok operacji; informacja o błędzie winna być wykorzystywana przez system;

- powinny istnieć programy kontroli, które w systemach wielomaszynowych/wieloprocesorowych wykonywane były by przez procesory nie zajęte w danej chwili programami użytkowymi;
- każdy procesor powinien dopuszczać możliwość takich sytuacji, kiedy to zadanie opracowywane przez niesprawne urządzenie ma dostęp do wszystkich tablic systemowych i zarządzających zadaniami, wykorzystywanych w tym urządzeniu. Urządzenie - odbiornik, któremu przekazuje się rezultaty decyzji powinno znać stan, w którym znajduje się niesprawne urządzenie, żeby ocenić moment, kiedy należy wziąć na siebie wykonywanie zadania.

Podstawowe połączenia dwóch maszyn cyfrowych serii MERA-300 jako podstawowy składnik rekonfiguralnych systemów wielomaszynowych

Połączenie dwóch maszyn będziemy realizować przez jednostkę łączącą (JL) opisaną w pracy [2]. Zapewnia ona wymianę informacji między pamięciami połączonych maszyn cyfrowych. Zawiera ona słowo stanu, informujące o stanie, w jakim znajduje się i ma możliwość wysyłania dodatkowych przerwania z P1/2 do P2/1 prócz przerwania związanych z samą transmisją. Dwie maszyny połączone jedną jednostką przedstawia rys. 1.



Rys. 1. Połączenie dwóch maszyn cyfrowych serii MERA-300

Budowa jednostki łączącej umożliwia stworzenie systemu dwumaszynowego o podwyższonej niezawodności w stosunku do systemu jednomaszynowego. Do tego celu należy zastosować środki sprzętowe i programowe. Minikomputery serii MERA-300 nie mają żadnych rozwiązań konstrukcyjnych umożliwiających wykrycie nieprawidłowej pracy jednostki centralnej, a mogą jedynie sygnalizować nieprawidłową pracę urządzeń zewnętrznym do nich dołączonych przez mechanizm słowa stanu. Jednostka łącząca spełnia również tę zasadę, przy czym rejestr słowa stanu jest dostępny programowo dla obu maszyn. Każda z nich może korzystać z tego rejestru niezależnie, w każdej chwili. Jednostka łącząca sygnalizuje w słowie stanu fakt zaniku napięcia zasilania do-

łączonej do niej maszyny. Jest to jedyna metoda, która informuje sąsiednią maszynę, że jej "PARTNER" jest niesprawny. Oprogramowanie ma zatem pełną informację o stanie jednostki łączącej, potrzebną do przeprowadzenia transmisji i oceny jej poprawności.

Oprogramowanie powinno realizować następujący algorytm wymiany informacji:

- jedna maszyna (nadawca) inicjuje transmisję do drugiej maszyny (odbiorcy) - jeśli jednostka jest sprawna,
- jeśli odbiorca jest "sprawny", to rozpoczyna się transmisja,
- po zakończeniu transmisji odbiorca sprawdza jej poprawność (np. na podstawie sumy kontrolnej, bitu parzystości itp.). Jeśli transmisja jest poprawna to odbiorca wysyła do nadawcy przerwanie, mające sens akceptacji transmisji, a w przeciwnym razie wysyła przerwanie informujące o błędnej transmisji. Jeśli transmisja jest błędna można ją powtórzyć kilkakrotnie.

W wypadku, kiedy niesprawne jest połączenie między maszynami oprogramowanie powinno wykrywać niesprawne urządzenie (JL, P1 lub P2). Do tego celu powinny służyć specjalne programy testujące jednostkę łączącą i oba procesory. Następnie system powinien zmienić swoją konfigurację (o ile jest to możliwe) w celu dalszego działania. Do zmiany konfiguracji służą dwie operacje jednostki łączącej "odłącz" i "podłącz", oznaczające logiczne odłączenie lub podłączenie maszyny do danej jednostki łączącej. System po rekonfiguracji może spełnić wszelkie poprzednie funkcje lub ich część w zależności od konkretnych zastosowań.

Zwiększenie niezawodności połączenia dwóch maszyn cyfrowych serii MERA-300 można uzyskać przez połączenie obu maszyn za pomocą więcej niż jednej jednostki łączącej. System może przy normalnej

pracy korzystała ze wszystkich ścieżek połączeń, jak też z ich części, a w wypadku błędu jednej z nich - przesuwać funkcje na inną. Najprostszą realizacją jest dołączenie dwóch jednostek łączących.

• Systemy wielu połączonych maszyn serii MERA-300

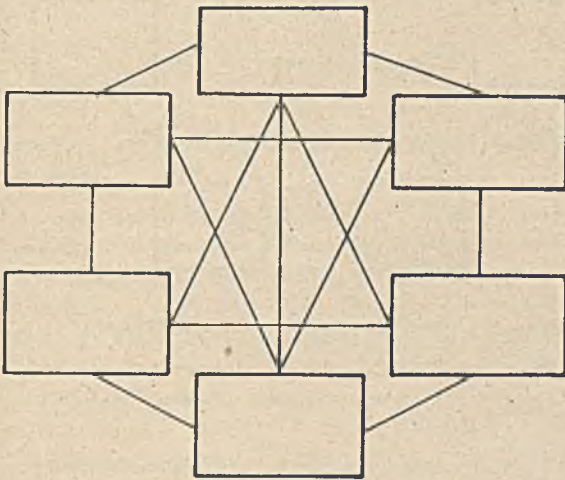
W pracy [2] omówiono szczegółowo różne konfiguracje połączonych maszyn cyfrowych serii MERA-300. Obecnie przedstawimy zagadnienia niezawodnościowe związane z tymi systemami, opierając się na rozważaniach o niezawodności dwóch połączonych maszyn cyfrowych, omówionych poprzednio. Zakładamy przy tym, że w każdej z maszyn cyfrowych znajduje się minimum oprogramowania niezbędne do zapewnienia testowania jednostki łączącej i samego procesora, a także umożliwiające zmianę jego konfiguracji.

• Kompletny układ połączeń - "każdy z każdym"

Układ ten (rys. 2) jest najkosztowniejszy, bowiem jeśli chcemy połączyć N maszyn to liczba jednostek łączących wyraża się wzorem:

$$I = \sum_{n=1}^N (n-1)$$

Dla minikomputera MERA-300 można połączyć maksymalnie 17 procesorów. Rozbudowa systemu jest kosztowna, bo prócz n -tego procesora należy dołączyć $n-1$ łączy.



Rys. 2. Kompletny układ połączeń
"każdy z każdym"

Zaletą tego układu są bardzo dobre parametry niezawodnościowe. Awaria jednego z procesorów nie powoduje awarii całego systemu, a tylko rekonfigurację systemu do podsystemu z kompletnym układem połączeń. Z punktu widzenia oprogramowania może być realizowany zarówno bezpieczny upadek, gdy jeden lub więcej procesorów w sieci są nadmiarowo oraz łagodny upadek, gdy pozostałe procesory przejmują funkcje uszkodzonego.

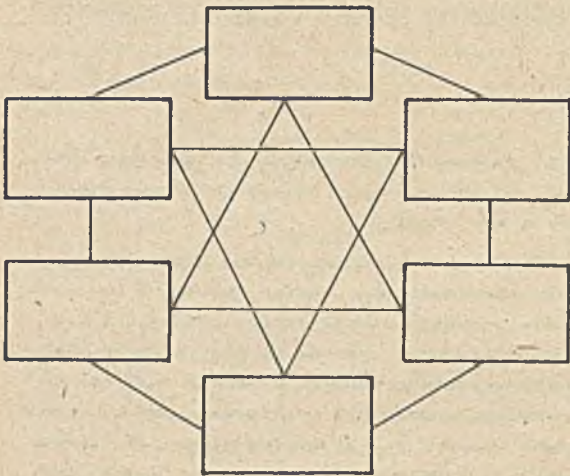
Mniej szkodliwe dla pracy systemu (mniejszy wpływ na efektywność niż przy awarii procesora) jest awaria jednostki łączącej, gdyż wówczas przerywana jest tylko jedna droga połączeń. W tej sytuacji system może zmienić konfigurację zarówno do podsystemu z kompletnym układem połączeń eliminując jeden procesor, jak i do dowolnej innej sieci połączeń, dającej się zrealizować. Wymagane jest wówczas, aby każdy z

procesorów miał tablicę odzwierciedlającą aktualny układ systemu. Przy takich założeniach awaria jednostki łączącej przerywa połączenie na drodze P_i, P_j i $i \neq j$, ale oba procesory mogą dalej komunikować się przez procesor P_k ($k \neq i, k \neq j$) lub inny. Można podwyższyć niezawodność systemu przez dublowanie niektórych lub wszystkich dróg połączeń. Ciągłe jednak słabym punktem pozostają pojedyncze procesory, dlatego też nie należy przydzielać konkretnego zadania tylko jednemu z procesorów. W wypadku awarii takiego pojedynczego procesora zadanie to nie będzie wykonane, bowiem dla maszyny cyfrowej z serii MERA-300 dostęp do informacji w pamięci możliwy jest wyłącznie przez jej procesor, zatem w wypadku jego uszkodzenia nie da się odtworzyć stanu programu sprzed awarii.

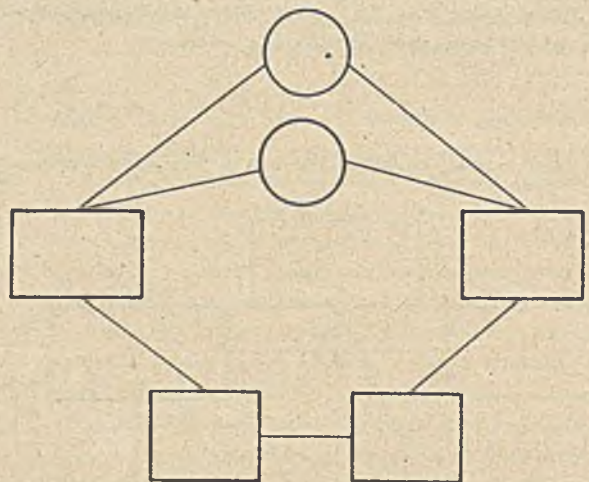
• Pętla

W systemie tym informacje z procesora P_i do P_j ($i \neq j$) przesyła się przez wszystkie pośrednie procesory - na ogół informacja krąży tylko w jednym konkretnym kierunku. Adresuje się zazwy-

czas procesy, a nie konkretne procesory. Podwyższenie niezawodności nastąpi, gdy zastosujemy dublowanie połączeń i połączenia omijające procesory (rys. 3). Wówczas awaria jednej jednostki łączącej spowoduje przesłanie informacji drugą jednostką łączącą. Awaria procesora spowoduje jego ominięcie (t.j. powstanie pętla o $N-1$ procesorach). W takim wypadku awaria systemu nastąpi tylko wówczas, gdy zdarzy się awaria dwóch kolejnych procesorów. Aby temu zapobiec można zastosować łącząca omijające dwa kolejne procesory – takie postępowanie prowadzi w końcu do zbudowania kompletnej sieci połączeń (rys. 2), omawianej poprzednio jednak o zupełnie innym oprogramowaniu. Podobne problemy niezawodnościowe występują w "pętli z centralnym przełącznikiem". Centralny przełącznik ma kluczowe znaczenie w przesyłaniu informacji i jego awaria przerywa pracę systemu (efektywność spada do zera), ponieważ nie można zastosować omijania procesora – "centralnego przełącznika". W tej sytuacji należy go zdublować czy nawet zwielokrotnić (rys. 3 a). Zagadnienia te są identyczne jak dla układu "gwiazda z centralnym przełącznikiem".



Rys. 3. Pętla o podwyższonej niezawodności



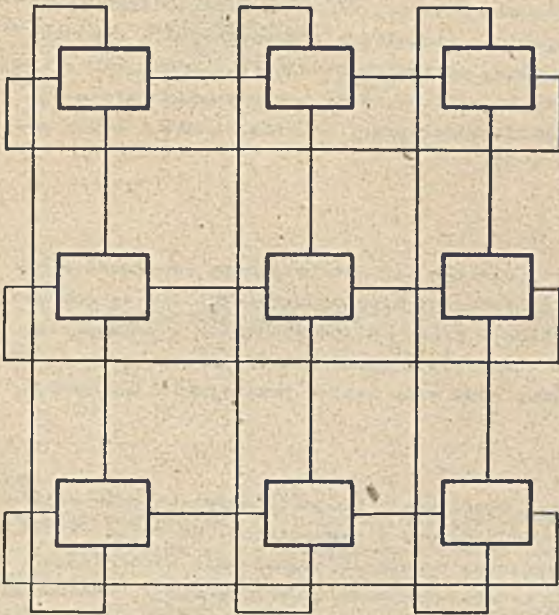
Rys. 3a. Pętla z dublowaniem centralnego przełącznika

• Regularna sieć

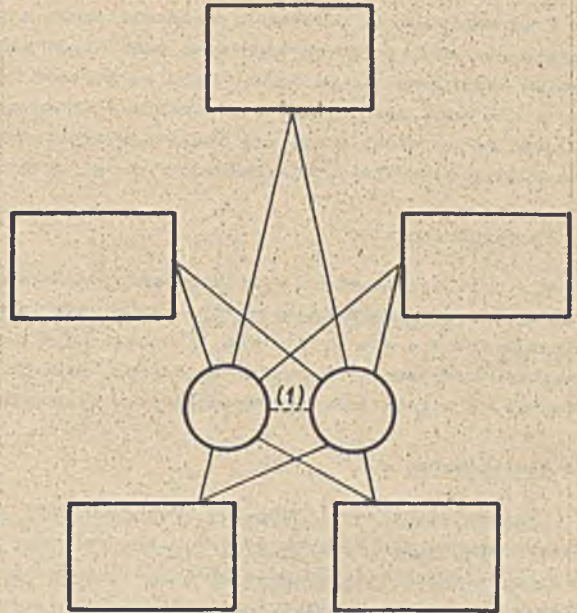
Sieć regularna jest tematem wielu prac teoretycznych, ale rzadko realizuje się ją w praktyce. Główną przeszkodą jest tu złożoność oprogramowania, które komplikuje się jeszcze bardziej przy spełnianiu wymagań niezawodnościowych. Awaria ścieżki łączącej nie jest wypadkiem najgorszym i ma mniejszy wpływ na efektywność systemu niż uszkodzenie jednego z procesorów, które dezorganizuje całą sieć. W takim wypadku konieczna jest rekonfiguracja sieci i zmiana algorytmu przesyłania informacji między procesorami. Sieć po rekonfiguracji powinna działać już nie jako sieć regularna, gdyż dla zapewnienia regularności należałoby wyłączyć z sieci więcej niż jeden (ten uszkodzony) procesor (np. dla sieci na rys. 4 należy wyłączyć 5 procesorów). Widać zatem, że każda awaria jednego z elementów przetwarzających powoduje wyłączenie z pracy znacznej jej części, co znacznie odbija się na efektywności systemu, zaś dążenie do utrzymania efektywności na niezmiennym poziomie prowadzi do zasadniczej zmiany organizacji systemu (sieć przestaje być siecią regularną).

• "Gwiazda z centralnym przełącznikiem"

Układ typu gwiazda jest wysoce zawodny, gdyż awaria elementu przełączającego powoduje awarię całego systemu. W celu podwyższenia jego niezawodności należałoby zdublować (nawet zwielokrotnić) element przełączający. Przykład rozwiązania dla minikomputerów serii MERA-300 podaje rys. 5. Zdublowany element przełączający może działać wg różnych algorytmów. Przy istniejącym połączeniu przez jednostkę łączącą (1), jeden z procesorów nie uczestniczy w przekazywaniu informacji, a je-



Rys. 4. Sieć regularna



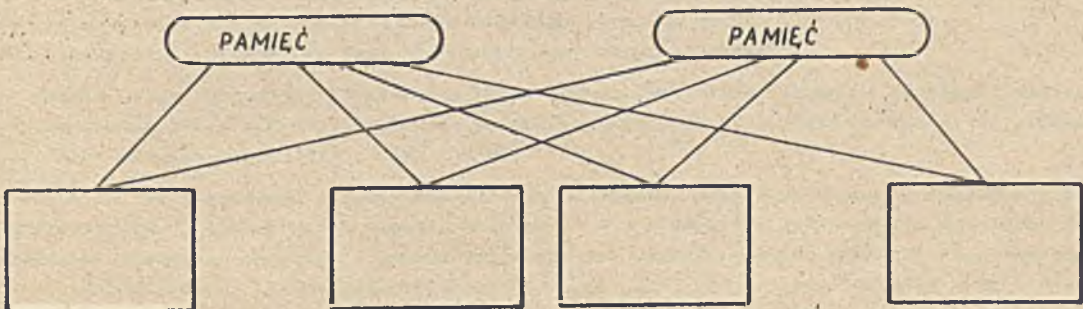
Rys. 5. Gwiazda z dublowaniem centralnego przełącznika

dynie testuje drugi, działający jako przełącznik. W wypadku wykrycia uszkodzenia przejmuje jego funkcje.

Przy zastosowaniu jednostki łączącej, użyteczne stają się operacje "ODŁACZ" i "PODLACZ", pozwalające zablokować połączenia z uszkodzonym elementem. Inny algorytm współpracy nie wykorzystujący tego połączenia realizuje przekazywanie informacji równocześnie przez obydwa elementy przełączające, a minikomputer-odbiorca porównuje obie informacje, które powinny być identyczne. System ten można jeszcze rozbudować realizując algorytm "głosowania 2 z 3", gdzie w sieci byłyby trzy elementy przełączające. Prawdziwa byłaby informacja identyczna przy transmisji przez co najmniej dwa elementy przełączające.

• Wspólna pamięć

Jak omówiono w pracy [2] funkcje "wspólnej pamięci" w systemie opartym na minikomputerach serii MERA-300 i jednostce łączącej spełnia jeden z minikomputerów. Z punktu widzenia efektywności układ ten jest identyczny z omawianym poprzednio, bowiem awaria wspólnej pamięci jest równoważna awarii całego systemu. Podobnie jak poprzednio zdublowanie pamięci jest jednym ze sposobów na zwiększenie niezawodności systemu (rys. 6).



Rys. 6. Układ "wspólna pamięć" o podwyższonej niezawodności

W odróżnieniu od rozwiązań przedstawionych w poprzednim punkcie, każdy ze zwielokrotnionych procesorów zawiadujących wspólnymi pamięciami musi pracować równolegle z pozostałymi w wypadku zapisu informacji do pamięci (chyba że pracują na wspólnej pamięci masowej). W wypadku rekonfiguracji nie może ulec zmianie informacja o aktualnym stanie systemu. Przy odczytywaniu informacji ze wspólnej pamięci staje się możliwe wykrycie nieprawidłowości pracy wspólnej pamięci przez porównywanie odczytywanej informacji (np. głosowanie 2 z 3).

• Nieregularna sieć

W sieci tej nie można wyróżnić żadnej regularności, dlatego też istnieje duża dowolność w rekonfiguracji systemu, a co za tym idzie sieć ma dobre własności niezawodnościowe. Nie są one jednak tak istotne dla tych sieci, ponieważ (jak wspomniano w pracy [2]) powstają one z połączenia wielkich stacjonarnych maszyn cyfrowych, pracujących w odległych ośrodkach obliczeniowych. Istotniejsze są zagadnienia niezawodnościowe połączeń między nimi oraz koszty przesyłania informacji.

Podsumowanie

Minikomputery serii MERA-300 oraz jednostka łącząca umożliwiają projektowanie systemów wielomaszynowych spełniających odpowiednie wymagania niezawodnościowe i ekonomiczne. Właściwe określenie wyjściowych efektów działania projektowanego systemu na podstawie odpowiednio wyróżnionych stanów pracy tego systemu, w połączeniu z możliwością wyboru odpowiedniej konfiguracji, dublowania łączy, maszyn i możliwości rekonfiguracji zapewnia dobre właściwości eksploatacyjne budowanego systemu.

Posiadanie odpowiedniej liczby jednostek łączących oraz minikomputerów serii MERA-300 umożliwia modelowanie systemów wielomaszynowych oraz badanie ich możliwości. Systemy takie są również bardzo przydatne w dydaktyce jako nieoceniona pomoc laboratoryjna przy nauczaniu przedmiotów związanych z organizacją, programowaniem i niezawodnością systemów łączących.

Literatura

- [1] ENSLOW P.H.: Multiprocessors and parallel processing. A. Wiley - Interscience Publication, 1974
- [2] KOWALEWSKI L., KUCHCIŃSKI K., WISZNIEWSKI D.: Systemy wielomaszynowe oparte na minikomputerach z serii MERA-300. Biuletyn Informacyjny Obiektowe Systemy Komputerowe, 1978 nr 1
- [3] PIERCE W.H.: Failure - tolerant computer design. New York: Academic Press, 1965
- [4] ZAMOJSKI W.: Zagadnienie niezawodności w procesie projektowania systemów. Prace Naukowe Instytutu Cybernetyki Technicznej Politechniki Wrocławskiej. Wrocław 1976

dr inż. Mieczysław MURASZKIEWICZ
Instytut Informacji Naukowej
Technicznej i Ekonomicznej

Realizacja mnożenia, wyszukiwania i sortowania za pomocą układów komórkowych

Wstęp

Układem komórkowym (cellular array, iterative array) nazywa się strukturalnie jednorodny urządzenie, które składa się z określonej liczby identycznych, wzajemnie połączonych w regularną sieć elementów nazywanych komórkami. Budowa wewnętrzna komórki oraz konfiguracja sieci zależą od przeznaczenia układu komórkowego. Układy komórkowe rozważane w niniejszym artykule są układami:

- makrokomórkowymi (komórki są złożone z kilkunastu do kilkudziesięciu bramek logicznych);
- kombinacyjno-sekwencyjnymi (komórki są układami logicznymi właśnie takiego typu);
- programowalnymi (istnieje możliwość wpływu na funkcje logiczne realizowane przez komórki, a także na strukturę połączeń między nimi).

Jednorodność struktury układów komórkowych decyduje o tym, że można je łatwo wytwarzać posługując się technologią obwodów scalonych. Obecny stan tej technologii pozwala upakować na powierzchni kilku milimetrów kwadratowych wiele tysięcy bramek logicznych przy opóźnieniu wnoszonym przez bramkę ok. 2-5 ns. Ograniczeniem wynikającym z małej objętości układów jest nieduża liczba doprowadzeń zewnętrznych.

Badania nad układami komórkowymi rozpoczęto w latach pięćdziesiątych. W 1961 r. F. Hannie opublikował klasyczną już dzisiaj pracę na temat tych układów i ich właściwości [6]. Dopiero jednak przełom lat sześćdziesiątych i siedemdziesiątych przyniósł szersze zainteresowanie układami komórkowymi wśród konstruktorów maszyn cyfrowych. Należy to tłumaczyć z jednej strony postępem technologii wytwarzania obwodów scalonych, a z drugiej strony - intensywnymi badaniami w zakresie organizacji maszyn cyfrowych.

W pracy [4] M. Flynn podał cztery zasadniczo typy organizacji maszyn cyfrowych w zależności od strumienia instrukcji i danych. Są to komputery typu:

- PIPD (pojedynczy strumień instrukcji, pojedynczy strumień danych) - do grupy tej należą maszyny o organizacji zaproponowanej przez von Neumanna,
- WIPD (wielokrotny strumień instrukcji, pojedynczy strumień danych),
- PIWD (pojedynczy strumień instrukcji, wielokrotny strumień danych),
- WIWD (wielokrotny strumień instrukcji, wielokrotny strumień danych).

Mimo różnic między jednostkami przetwarzającymi rozważanymi przez Flynn'a, a układami komórkowymi, zaproponowaną przez niego klasyfikację można zastosować do tych układów. Układy te pozwalają zrealizować jednostki typu PIWD, WIPD i WIWD.

W pracach nad zastosowaniem układów komórkowych szczególny nacisk położono na realizację operacji arytmetycznych, takich jak mnożenie i dzielenie [2]. Pracom tym towarzyszyła nadzieja za projektowania urządzenia lepszego - przede wszystkim pod względem czasu wykonywania wymienionych operacji - od stosowanych powszechnie arytmetrów. Następne lata nie potwierdziły całkowicie tych nadziei, choć podjęto próby zwiększenia efektywności układów komórkowych, przez przystosowanie ich do potokowego (pipeline) przetwarzania danych. Zwiększało to niekiedy efektywność

o 230 % [5]. Pojawily się również prace wskazujące sposoby użycia układów komórkowych do realizacji operacji nienumerycznych, takich jak przechowywanie, wyszukiwanie i sortowanie danych [8], [10], [11]. Poniższe informacje o nurtach zastosowań układów komórkowych zawiera praca [3].

Zainteresowanie układami komórkowymi nie słabnie. Tłumaczy się to ich licznymi zaletami, do których należą m.in.:

- jednorodność struktury,
- samosterowalność,
- stosunkowo niska cena,
- duża niezawodność,
- łatwość wytwarzania,
- uniwersalność zastosowań.

Za zaletę uważa się to, że w układach komórkowych program działania określony jest przez konfigurację sprzętu, a nie przez ciąg rozkazów tak, jak w klasycznym komputerze. Fakt ten przybliża możliwość realizacji idei zastąpienia oprogramowania sprzętem. Ponadto wydaje się, że zastosowanie układów komórkowych do operacji nienumerycznych może przybliżyć urzeczywistnienie myśli Johna von Neumanna o utworzeniu jednorodnego funkcjonalnie urządzenia przydatnego do jednoczesnego przechowywania i przetwarzania danych bez konieczności ich przesyłania z pamięci do arytmometru, a przy tym umożliwiające równoległe wykonywanie wielu działań.

Poniżej będą przedstawione zasady wykonywania mnożenia, przechowywania, wyszukiwania i sortowania za pomocą układów komórkowych. Dla uproszczenia wyjaśnień przyjmuje się, że mnożenie i sortowanie dotyczy liczb dodatnich przedstawionych w naturalnym zapisie dwójkowym (bez znaku). Podane niżej zasady wyszukiwania i sortowania oparto na wynikach uzyskanych w pracy [10].

• Mnożenie

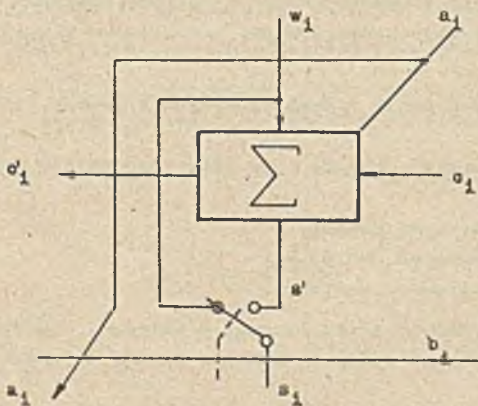
Mnożenie dwóch liczb można w najprostszym sposobie wykonać przez wyznaczanie kolejnych iloczynów częściowych powstałych z dodawania odpowiednio przesuwanej mnożnej. Np. dla $a=7$ (dwójkowo: 111), $b=6$ (dwójkowo: 110) iloczyn $c=a \times b = 7 \times 6 = 42$ (dwójkowo 101010) mnożenie przebiega w następujący sposób:

| | | |
|---|-------------|--|
| a | 1 1 1 | |
| b | 1 1 0 | |
| | 0 0 0 | |
| | 1 1 1 | |
| | 1 1 1 | |
| c | 1 0 1 0 1 0 | |

iloczyn częściowy

0 0 0
0 0 0
1 1 1

Układ komórkowy do realizacji mnożenia można opisać na tzw. komórce Hoffmana [7]. Schemat funkcjonalny tej komórki podano na rys. 1. Wejściami komórki są: bit mnożnej a_1 , bit mnożnika b_1 , bit iloczynu częściowego w_1 oraz bit przeniesienia z poprzedniej komórki c_1 . Wyjściami komórki są: bit sumy s_1 , powtarzane z wejścia bity a_1 , b_1 oraz bit przeniesienia do następnej komórki c_1 . W skład komórki wchodzi układ sumatora jednobitowego, który na rys. 1 oznaczono symbolem Σ oraz przełącznik dwustanowy (w praktyce jest on zrealizowany za pomocą bramek logicznych), którego położenie zależy od wartości bitu b_1 . Działanie komórki Hoffmana można opisać w następujący sposób:

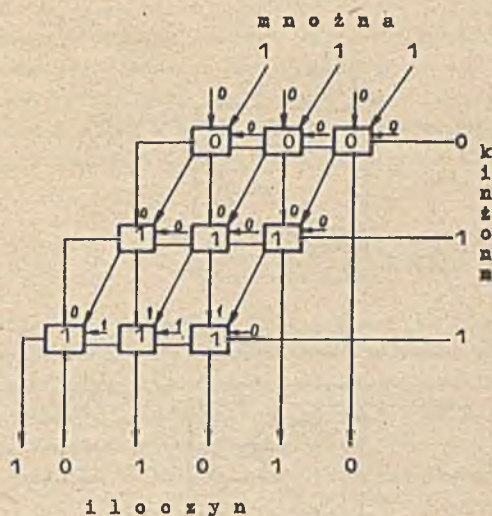


Rys. 1. Schemat komórki Hoffmana

$$s_1 = \begin{cases} w_1, & \text{dla } b_1 = 0 \\ a_1, & \text{dla } b_1 = 1 \end{cases} \quad (1)$$

A zatem, gdy bit mnożnika b_1 ma wartość 1 komórka wykonuje operację dodawania jednobitowego, natomiast gdy $b_1 = 0$ - komórka przesyła bit iloczynem częściowym w_1 ze swego wejścia na wyjście bez żadnych zmian.

Na rys. 2 przedstawiono schemat układu komórkowego, który realizuje mnożenie dwóch liczb trzybitowych. Liczby występujące na rysunku są takie same jak w podanym wyżej przykładzie.



Rys. 2. Przykład mnożenia za pomocą układu komórkowego

Każdemu bitowi mnożnika jest przyporządkowany jeden poziom układu komórkowego złożony z trzech połączonych szeregowo komórek. Komórki te realizują dodawanie umożliwiając w rezultacie wyznaczenie kolejnego iloczynu częściowego. Dla polepszenia czytelności, na rys.2 w kwadratach oznaczających komórki wpisano bity tworzące liczby dwójkowe, które po zsumowaniu dają iloczyn w podanym wyżej przykładzie mnożenia. Przeniesienia występujące między komórkami jednego poziomu zaznaczono symbolicznie na rys.2 skierowanymi liniami poziomymi, przy których podano wartości tych przeniesień. Pozostałe linie poziome oznaczają drogi przesłania bitów mnożnika. Pionowe linie skierowane, łączące komórki, oznaczają drogi przesłania bitów iloczynów częściowych, a linie ukośne - drogi przesłania bitów mnożnej. Przyjmując, że czas opóźnienia wnoszony przez komórkę ma wartość t , oraz że układ komórkowy jest przeznaczony do mnożenia dwóch n -bitowych

liczb binarnych, a więc składa się z $n \times n$ komórek, można łatwo wykazać, że czas wykonania tej operacji wynosi około $2nt$.

Jak już wspomniano we wstępie, układy komórkowe zastosowano także do realizacji dzielenia. Budowa takich układów jest bardziej złożona niż dla mnożenia [2]. W najprostszym wypadku dzielenie za pomocą układu komórkowego - podobnie jak dla mnożenia - jest niemal dokładnym odwzorowaniem tej operacji realizowanej przez człowieka za pomocą "ołówka i papieru". Znane są uniwersalne układy macierzowe, w których za cenę rozbudowy komórek, zwiększenia ich liczby i wprowadzenia dodatkowego układu sterującego można wykonywać zarówno dzielenie jak i mnożenie [1].

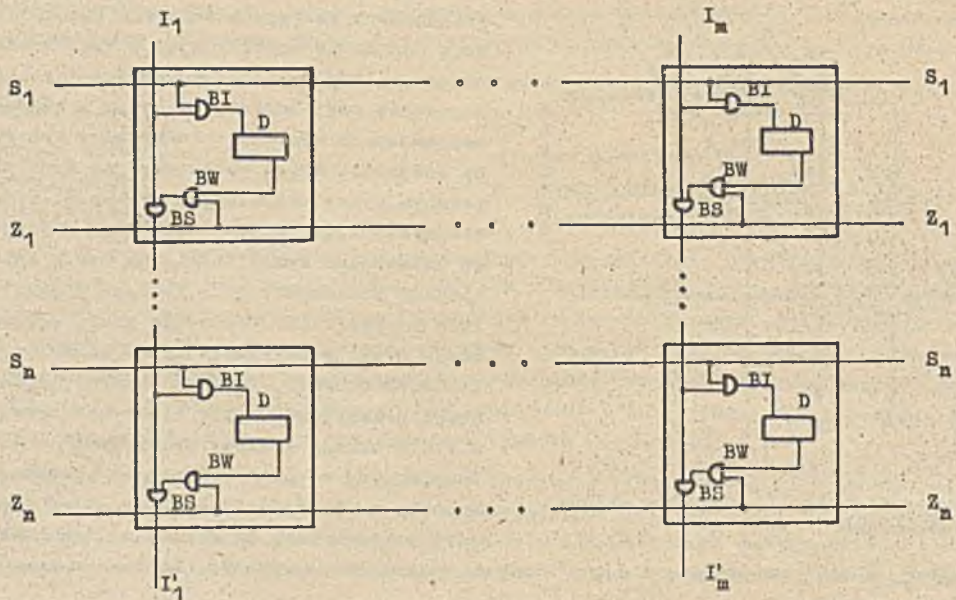
• Przechowywanie

Ważnym udoskonaleniem rozszerzającym zakres zastosowań układów komórkowych jest wyposażenie ich w zdolność przechowywania danych. W tym celu do każdej komórki dołącza się przerzutnik, który służy do przechowywania jednego bitu, oraz dodatkowe bramki logiczne spełniające funkcje pomocnicze.

Na rys.3 przedstawiono przykład układu komórkowego umożliwiającego przechowywanie danych. Komórki tego układu połączone są w taki sposób, że tworzą tablicę prostokątną złożoną z n wierszy i m kolumn. W każdym wierszu tej tablicy przechowywane jest jedno słowo binarne. Dla zwiększenia przejrzystości rysunku uwidocznił na nim tylko elementy związane z przechowywaniem, wprowadzaniem i odczytywaniem danych. Wprowadzanie bitu do komórki znajdującej się na przecięciu i -tego wiersza i j -tej kolumny układu komórkowego odbywa się przez podanie go do szyny I_j przy jednoczesnym podaniu sygnału wprowadzenia do szyny S_i , co "otwiera" bramkę iloczynu logicznego BI, a w konsekwencji umożliwia zapis do przerzutnika (na rys.3 jest to przerzutnik typu D). Odczyt informacji z i -tego wiersza odbywa się przez podanie sygnału odczytu do szyny Z_i , co "otwiera" bramkę iloczynu logicznego BW. Bramka BS jest bramką sumy logicznej. Należy zwrócić uwagę, że w danej chwili możliwy jest odczyt tylko jednego słowa. Odczytane bity otrzymuje się z szyn pionowych I'_1, I'_2, \dots, I'_m . Można zauważyć, że przedstawiony układ komórkowy pozwala na przepisywanie danych ze wskazanego wiersza do innego dowolnego wiersza. Możliwe jest także przepisywanie pojedynczych bitów.

• Wyszukiwanie

Niech układ komórkowy składa się z $n \times m$ komórek tworzących tablicę prostokątną. Tablica ta składa się z n wierszy numerowanych od 1 do n poczynając od "góry" oraz z m kolumn numerowanych od 1 do m licząc od strony lewej do prawej strony tablicy. Każda komórka jest połączona ze wszystkimi komórkami sąsiadującymi z nią bezpośrednio w wierszu i w kolumnie. Na rys. 4 podano schemat

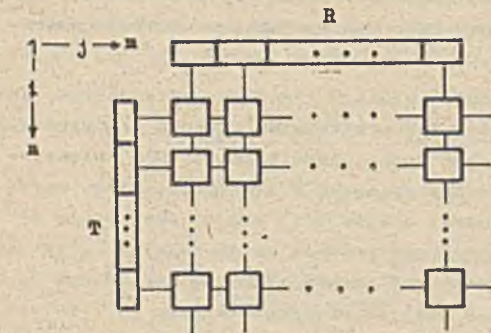


Rys. 3. Schemat układu komórkowego do przechowywania danych

rozważanego układu komórkowego. W dalszym ciągu układ ten będzie nazywany macierzą. Kwadraty na rys. 4 oznaczają komórki, a linie łączące kwadraty symbolizują połączenia występujące między komórkami. Blok T oznacza tzw. rejestr przeniesień, zaś blok R - tzw. rejestr wyszukiwawczy. W celu uproszczenia opisu na rys. 4 nie zaznaczono

sygnałów o stałej wartości dołączonych do komórek "brzegowych" macierzy.

Każda komórka jest wyposażona w jeden przerzutnik (element pamiętający), bramki logiczne tworzące układ umożliwiający porównanie dwóch bitów oraz bramki logiczne i linie transmisji pozwalające wpisywać i odczytywać bit do/z komórki (podobnie jak w przykładzie przedstawionym na rys. 3). Bit przechowywany w przerzutniku jest nazywany zawartością komórki. Komórkę, która znajduje się na przecięciu i -tego wiersza i j -tej kolumny dla $i \in \langle 1, n \rangle$, $j \in \langle 1, m \rangle$ oznacza się przez $K(i, j)$. Zawartość komórki $K(i, j)$ oznacza się przez $z(i, j)$. W wierszach macierzy są przechowywane słowa binarne. Słowo znajdujące się w i -tym wierszu macierzy będzie oznaczane przez r_i , przy czym $r_i = z(i, 1)z(i, 2) \dots z(i, m)$.



Rys. 4. Schemat układu komórkowego macierzy do wyszukiwania

Za bardziej znaczące pozycje słowa uważa się te, które znajdują się po jego lewej stronie.

Komórka może wykonać działanie, nazywane dalej porównaniem bitów, które polega na zrealizowaniu następującej funkcji $q(i, j)$, nazywanej przeniesieniem q dla $i = 1, 2, \dots, n$; $j = 1, 2, \dots, m$:

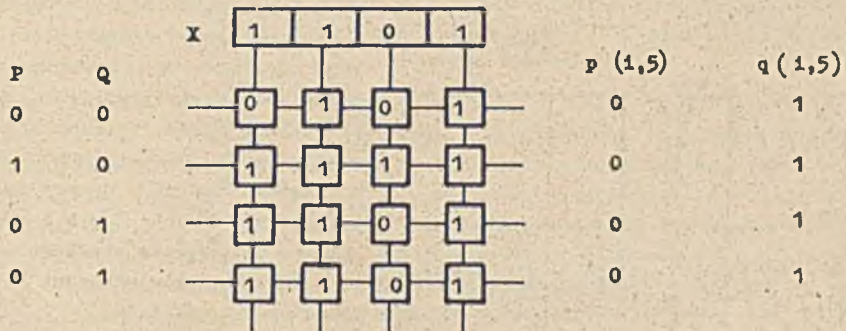
$$q(i, j) = \begin{cases} 0, & \text{gdy } b_j \neq z(i, j) \vee q(i, j+1) = 0 \\ 1, & \text{w przeciwnym razie} \end{cases} \quad (2)$$

$$q(i, m+1) = 1$$

gdzie b_j jest bitem należącym do słowa $x = b_1 b_2 \dots b_m$, które znajduje się w rejestrze R. Porównanie bitów pozwala ustalić, czy dwa bity są w relacji $=$, czy w relacji \neq , przy czym podczas porównania w komórce $K(i, j)$ uwzględnia się przeniesienie q , którym jest wynik porównania uzyskany w komórce $K(i, j+1)$. Biorąc pod uwagę budowę macierzy można pokazać, że szeregowe wykonanie m porównań bitów w kolejnych komórkach wiersza poczynając od komórki $K(i, m)$, a kończąc na komórce $K(i, 1)$ powoduje wygenerowanie przeniesienia $q(i, 1)$ o wartości określonej w następujący sposób:

$$q(i, 1) = \begin{cases} 1, & \text{gdy } x = r_i \\ 0, & \text{gdy } x \neq r_i \end{cases} \quad (3)$$

Oznacza to, że możliwa jest realizacja wyszukiwania, przy czym słowo x jest kluczem, według którego prowadzi się wyszukiwanie (jest ono przechowywane w rejestrze wyszukiwawczym R), zaś słowo r_i jest daną, która podlega wyszukiwaniu. Łatwo spostrzec, że wyszukiwanie można wykonywać jednocześnie w odniesieniu do wszystkich słów r_i dla $i = 1, 2, \dots, n$ umieszczonych w wierszach macierzy. Wynikiem wyszukiwania jest słowo $Q = q(1, 1)q(2, 1) \dots q(n, 1)$ stanowiące zawartość rejestru przeniesień T. Na rys. przedstawiono macierz dla $n = m = 4$. W wierszach macierzy przechowywane są



Rys. 5. Przykład wykonania operacji porównania

następujące słowa: $r_1 = 0101$, $r_2 = 1111$, $r_3 = r_4 = 1101$ - odpowiednie bity słów wpisano na rysunku w kwadraty symbolizujące komórki. Na rysunku uwidoczniono słowo $x = 1101$ oraz przeniesienia $q(i, 5)$ dla $i = 1, 2, 3, 4$. Słowo Q utworzone z tych przeniesień znajduje się w lewej części rysunku.

Rozszerzenie możliwości macierzy można uzyskać przez rozbudowę komórek. Nie ma przeszkód, aby komórka realizowała następującą funkcję $p(i, j)$, nazywaną przeniesieniem p , dla $i = 1, 2, \dots, n$; $j = 1, 2, \dots, m$:

$$p(i, j) = \begin{cases} 0, & \text{gdy } x_j = x(i, j) \wedge p(i, j+1) = 0 \vee \\ & \vee x_j = z(i, j) \\ 1, & \text{w przeciwnym razie} \end{cases} \quad (4)$$

$$p(i, m+1) = 0$$

Szeregowe wykonanie działań określonych przez formułę (4) w sposób analogiczny do opisanego wyżej porównania pozwala ustalić czy słowa są w relacji \geq , czy w relacji $<$, przy czym:

$$p(i, 1) = \begin{cases} 0, & x \geq r_i \\ 1, & x < r_i \end{cases} \quad (5)$$

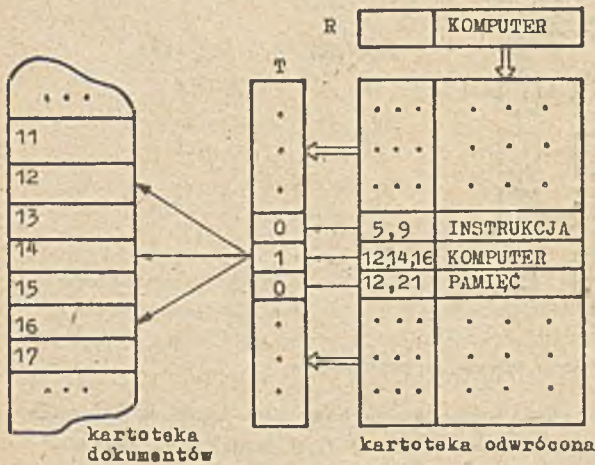
Wynikiem tych działań jest słowo $P = p(1, 1)p(2, 1) \dots p(n, 1)$ stanowiące zawartość rejestru przeniesień T. Na rys. 5 podano zawartość tego rejestru dla przedstawionej na nim macierzy.

Należy zwrócić uwagę, że macierz umożliwia wyszukiwanie jednoczesne, a zatem czas wyszukiwania nie zależy od mocy przeszukiwanego zbioru, a jedynie od długości słów. Macierz jest w tym wypadku

zarazem pamięcią, którą można zaliczyć do grupy pamięci asocjacyjnych.

Uogólnieniem przedstawionych w tym i poprzednim punkcie rozważań jest przyjęcie założenia, że zawartością komórki nie jest bit lecz znak, którym może być cyfra dziesiętna, litera alfabetu łacińskiego itp. W praktyce oznacza to rozbudowę komórki lub traktowanie kilku komórek "bitowych" jako jednej komórki "znakowej". Dalszym rozszerzeniem rozważań jest dopuszczenie maskowania wskazanych pozycji słów, co technicznie oznacza wyłączenie z działania tych kolumn macierzy, które odpowiadają zamaskowanym pozycjom.

Podane wyżej rozszerzenia pozwalają zastosować macierz do realizacji tzw. kartoteki odwróconej, która jest szczególnie użyteczna w systemach informacyjno-wyszukiwawczych o krótkim czasie udzielania odpowiedzi na pytania [9]. W tym wypadku do poszczególnych wierszy macierzy wpisuje się słowa, z których każde składa się z dwóch części. Część pierwsza zawiera słowo informacyjne, a część druga - listę adresów (lub numerów) dokumentów zawierających w swej treści to słowo informacyjne. Dokumenty są przechowywane w klasycznie zorganizowanej (np. sekwencyjnie) kartotece. W czasie wyszukiwania fragmenty słów zawierające listy adresów są zamaskowane. Adresy te będą użyte po wykonaniu operacji na macierzy w celu wskazania dokumentów w kartotece. Ilustracją przedstawionego sposobu zastosowania macierzy do realizacji kartoteki odwróconej jest kartoteka przedstawiona na rys.6. Pozycje mniej znaczące słów służą do przechowywania deskryptorów. Na rysunku uwidoczniło tylko trzy deskryptory: instrukcja, komputer, pamięć.



Rys. 6. Realizacja kartoteki odwróconej za pomocą układu komórkowego

komputer, pamięć. Deskryptor, który jest kluczem wyszukiwawczym (zwykle jest to element pytania wyszukiwawczego) umieszczono w rejestrze R. Pozycje słowa, zarezerwowane na listę numerów, zawierają numery dokumentów w kartotece, w których występuje deskryptor podany na pozostałych pozycjach. Z rys.6 wynika, że w przypadku wyszukiwania opartego na operacji badania równości słów wskazane będą dokumenty o numerach 12, 14 i 16.

• Sortowanie

Przyjmuje się, że do sortowania (porządkowania) będzie użyty macierz o takiej samej strukturze jak macierz opisana wyżej. Komórka $K(i,j)$ realizuje dla $i = 2,3,\dots,n$; $j = 1,2,\dots,m$ następującą funkcję:

$$p'(i,j) = \begin{cases} 0, & \text{gdy } z(i-1,j) = z(i,j) \wedge p'(i,j+1) = 0 \vee \\ & \vee z(i-1,j) > z(i,j), \\ 1, & \text{w przeciwnym razie} \end{cases} \quad (6)$$

$$p'(i,m+1) = 0$$

gdzie $p'(i,j)$ i $z(i,j)$ mają analogiczny sens jak dla zależności (4). Można pokazać, że szeregowe wykonanie działań określonych przez formułę (6) w sposób podobny do już poprzednio opisanego porównania, pozwala ustalić czy słowa sąsiadujące ze sobą w wierszach macierzy są w relacji \geq , czy w relacji $<$, przy czym:

$$p'(i,1) = \begin{cases} 0, & \text{gdy } r_{i-1} \geq r_i \\ 1, & \text{gdy } r_{i-1} < r_i \end{cases} \quad (7)$$

dla $i = 2,3,\dots,n$. Ponadto zakłada się, że komórka może wymieniać swoją zawartość z zawartością komórki sąsiadującej z nią w kolumnie. Słowa podlegające sortowaniu są przechowywane w wierszach macierzy. Ich kolejność jest dowolna. Z macierzą stowarzyszony jest układ sterujący, którego częścią jest rejestr przeniesień T. Układ ten składa się z n komórek - każda stowarzyszona jest z jednym wierszem macierzy.

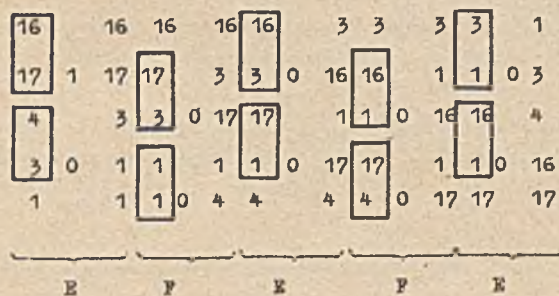
Sortowanie ciągu n słów przechowywanych w macierzy polega na wykonaniu n tzw. sekwencji. Różnią się dwa rodzaje sekwencji: parzystą, oznaczaną przez E i nieparzystą, oznaczaną przez F . Działania realizowane w ramach sekwencji E są następujące:

- we wszystkich wierszach o numerach parzystych^{*)} realizuje się jednocześnie wyznaczanie bitów $p'(i,1)$ zgodnie z zależnością (6); bity przeniesień wygenerowane z najbardziej znaczących pozycji wszystkich wierszy parzystych są wpisywane do odpowiednich pozycji rejestru T należącego do układu sterującego; liczba porównań realizowanych jednocześnie w macierzy wynosi $\left\lfloor \frac{n}{2} \right\rfloor$;
- we wszystkich wierszach parzystych, dla których bity $p'(i,1)$ wpisane do rejestru T mają wartość 0, realizuje się jednocześnie operacje wymiany. Operacja ta polega na przepisanie słowa i -tego wiersza parzystego do $i-1$ -go wiersza nieparzystego, a równocześnie przepisaniem słowa z wiersza o numerze $i-1$ do wiersza o numerze i - sąsiadujące słowa są zatem wymienione. Operacją wymiany kieruje układ sterujący. Wykonanie sekwencji E polega więc na utworzeniu rozłącznych par słów, takich, że pierwszy element pary jest słowem z wiersza nieparzystego, a drugi element - słowem z wiersza parzystego. Gdy pierwsze słowo w parze jest większe lub równo drugiemu słowu, są one zamieniane miejscami; w przeciwnym razie położenie słów nie zmienia się.

Działania składające się na sekwencję F są analogiczne do działań tworzących sekwencję E z tą różnicą, że pierwszym elementem tworzonych par są słowa znajdujące się w wierszach parzystych, a drugim elementem - słowa z wierszy nieparzystych. Operacja wyznaczania bitów $p'(i,1)$ dotyczy wierszy nieparzystych, poczynając od trzeciego wiersza.

Można wykazać [10], że wykonanie n sekwencji według schematu: E, F, E, F, \dots lub schematu F, E, F, E, \dots , gdzie n jest liczbą elementów porządkowanego ciągu, prowadzi do przekształcenia ciągu w ciąg niemalejący. Ilustracją tego stwierdzenia jest rys. 7, na którym schematycznie przedstawiono przebieg przekształcania ciągu 16, 17, 4, 3, 1 przez kolejne wykonanie sekwencji E, F, E, F, E . Pierwsza kolumna od lewej w każdej sekwencji zawiera elementy przechowywane w kolejnych wierszach macierzy - rozważane pary wpisano w prostokąty. W kolumnie drugiej podano wartości bitów $p'(i,1)$, a w kolumnie trzeciej - ciąg przekształcony przez działanie sekwencji. Z rys. 7 wynika, że ciąg otrzymany jako wynik ostatniego przekształcenia jest ciągiem rosnącym.

Warto zauważyć, że przedstawiony sposób sortowania odznacza się tym, że czas sortowania jest równy nt , gdzie t jest czasem wykonania sekwencji. Zależy on więc liniowo od liczby porządkowanych elementów. Poza tym sposób ten nie wymaga użycia miejsc roboczych pamięci. Żadna ze znanych tradycyjnych metod sortowania nie charakteryzuje się tak korzystnymi parametrami.



Rys. 7. Przykład sortowania ciągu: 16, 17, 4, 3, 1

^{*)} dalej zamiast określenia "wiersz o numerze parzystym" będzie używane określenie "wiersz parzysty" (analogicznie: wiersz nieparzysty).

• Zakończenie

Przedstawione wyżej układy komórkowe odznaczające się regularnością budowy pozwalają realizować różnorodne operacje przetwarzania danych przedstawionych w postaci ciągów znaków o skończonej długości. Mogą to być zarówno operacje arytmetyczne jak i operacje innego typu. Możliwość ta jest wyjątkowo cenna z punktu widzenia konstruktorów projektujących uniwersalne urządzenia do przetwarzania danych.

Zastosowanie układów macierzowych mogłoby wprowadzić pewne zmiany funkcjonalne w urządzeniach do przetwarzania danych - ten sam układ może służyć zarówno do przechowywania danych jak i do ich przetwarzania, na przykład wyszukiwania i sortowania. Byłoby to więc urzeczywistnienie idei skonstruowania urządzenia do jednoczesnego przechowywania i aktywnego przetwarzania danych. Jest to bardzo obiecująca perspektywa, zwłaszcza, że większość operacji przetwarzania danych w obecnie stosowanych systemach informatycznych to wyszukiwanie, sortowanie i aktualizowanie zbiorów danych.

Literatura

- [1] Deegan I.: Concise cellular array for multiplication and division. Electronics Letters 1974 nr 7 s.720-724
- [2] Deverell J.: The design of cellular arrays for arithmetic. The Radio and Electronic Engineer 1974 vol.44 nr 1 s.21-26
- [3] Dędyś-Gajkowiak D.: Sieci komórkowe. Warszawa: COPAN 1973 nr 110
- [4] Flynn M.J.: Very high-speed computing systems. Proc.IEEE 1966 s.1901-1909
- [5] Hallin T.G., Flynn M.J.: Pipelining of arithmetic functions. IEEE Trans. on Computers 1972 C-21 nr 8 s.880-886
- [6] Henne F.S.: Iterative arrays of logical circuits. New York: John Wiley 1961
- [7] Hoffman J.C., Lacaze G., Csillag P.: Iterative logical network for parallel multiplication. Electronics Letters 1968 nr 4 s.168
- [8] Kantz W.H.: Cellular logic-in-memory array. IEEE Trans. on Computers 1969 C-18 nr 8 s.719-727
- [9] Meadow C.: Analiza systemów informacyjnych. Warszawa: WNT 1972
- [10] Muraszkiewicz M.: Algorytmy procedur nienumerycznych w procesach przetwarzania danych i sposoby ich realizacji w macierzowej jednostce funkcjonalnej. Praca doktorska, IPI PAN 1978
- [11] Ramamoorthy C.V., Turner J.L., Wah B.W.: A design of a fast cellular associative memory for ordered retrieval. IEEE Trans. on Computers 1978 C-27 nr 9 s.800-813

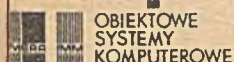
BRANŻOWY OŚRODEK INFORMACJI NAUKOWEJ TECHNICZNEJ I EKONOMICZNEJ
INSTYTUTU MASZYN MATEMATYCZNYCH
02-078 Warszawa, ul. Krzywickiego 34, tel. 21-84-41 w. 391

BOINTE udziela informacji

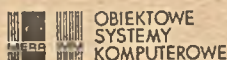
z zakresu techniki komputerowej

BOINTE wydaje

informacja ekspresowa

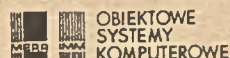


przegląd dokumentacyjny



Materiały konferencyjne, szkoleniowe, prospekty

biuletyn informacyjny



BOINTE gromadzi

wydawnictwa zwarte, czasopisma krajowe i zagraniczne, katalogi i prospekty, sprawozdania z prac naukowo-badawczych oraz inne materiały informacyjne

BOINTE wykonuje usługi reprodukcyjne i poligraficzne

fotokopie, mikrofilmy, kserokopie z zakresu posiadanych zbiorów

WARUNKI PRENUMERATY

Prenumeratę na kraj przyjmują Oddziały RSW "Prasa-Książka-Ruch" oraz urzędy pocztowe i doręczyciele w terminie do dnia 25 listopada na rok następny.

Cena prenumeraty rocznej zł 840.

Jednostki gospodarki uspołecznionej, instytucje, organizacje i wszelkiego rodzaju zakłady pracy zamawiają prenumeratę w miejscowych Oddziałach RSW "Prasa-Książka-Ruch", w miejscowościach zaś, w których nie ma Oddziałów RSW - w urzędach pocztowych.

Czytelnicy indywidualni opłacają prenumeratę wyłącznie w urzędach pocztowych i u doręczycieli.

Prenumeratę ze zleceniem wysyłki za granicę przyjmuje RSW "Prasa-Książka-Ruch", Centrala Kolportażu Prasy i Wydawnictw, ul. Towarowa 28, 00-958 Warszawa, konto PKO Nr 1153-201045.

Prenumerata ze zleceniem wysyłki za granicę jest droższa od prenumeraty krajowej o 50% dla zlecniodawców indywidualnych i o 100% dla zlecających instytucji i zakładów pracy.